hp 39gs and hp 40gs graphing calculators

Mastering the hp 39gs & hp 40gs

A guide for teachers, students and other users of the hp 39gs & hp 40gs

Edition 1.0

HP part number F2224-90010

Notice

REGISTER YOUR PRODUCT AT: www.register.hp.com

Hewlett-Packard Company
16399 West Bernardo Drive, MS 8-600
San Diego, CA 92123
USA

Acknowledgements

Hewlett-Packard would like to thank the author Colin Croft.

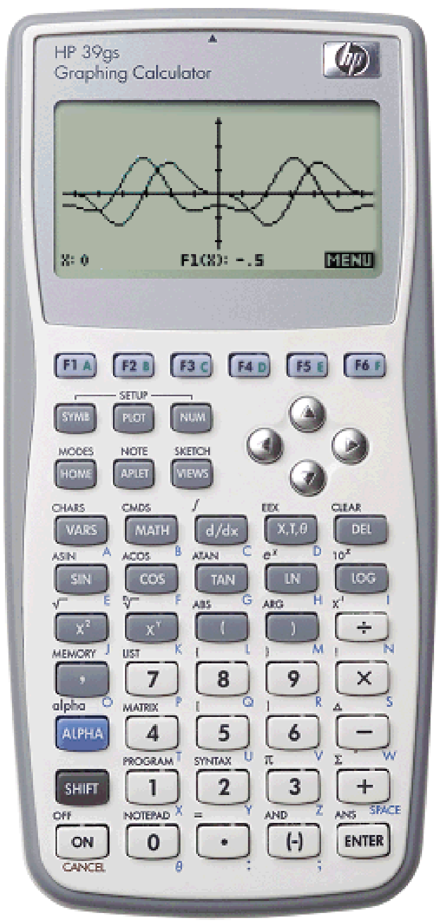
Printing History
Edition 1 December 2006

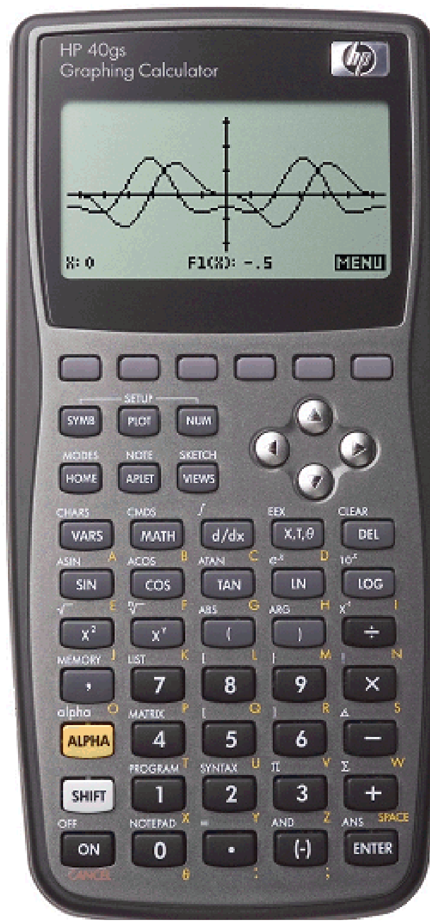
Table of Contents

2

INTRODUCTION

This book is intended to help you to master your hp 39gs or hp 40gs calculator but will also be useful to users of earlier models such as the hp 39g, hp 40g and hp 39g+. These are very sophisticated calculators, having more capabilities than a mainframe computer of the 1970s, so you should not expect to become an expert in one or two sessions. However, if you persevere you will gain efficiency and confidence.





The hp 39gs and hp 40gs, shown above, are 'sister' calculators released in 2006. They are identical in almost all respects except for their color schemes and in whether they have infra-red or a CAS. The hp 39gs was released mainly in the United States and other regions, such as Australia, which do not allow a Computer Algebra System, or CAS, in their educational systems. The hp 40gs, on the other hand, was released mainly in Europe where a CAS has long been an expected ability for calculators used by high school students. The hp 39gs has infra-red communication, similar to that of a TV remote control, which allows easy transmission of programs and aplets between calculators. The hp 40gs does not and uses a cable instead.

Many of the markets targeted by the hp 40gs do not allow infra-red communication in assessments and so, on the hp 40gs, this ability is permanently disabled, substituting instead a mini-serial cable supplied with the calculator (see page 237). The previous models, the hp 39g & hp 40g shared a common chip and, although it was never intended to be possible, a hacker released a special aplet for the hp 39g which would 'convert' it into an hp 40g and activate the CAS. This is not the case with the hp 39gs & hp 40gs: the internal chips are different and there is no way to 'convert' one into the other using an aplet or program.

For more information on the CAS, see page 324. This manual will cover, for the most part, the features which are shared by both calculators with the CAS covered in Appendix C. A detailed manual for the CAS is also supplied with the hp 40gs and more information can be found on the official HP website and on the author's website *The HP HOME view* at http://www.hphomeview.com.

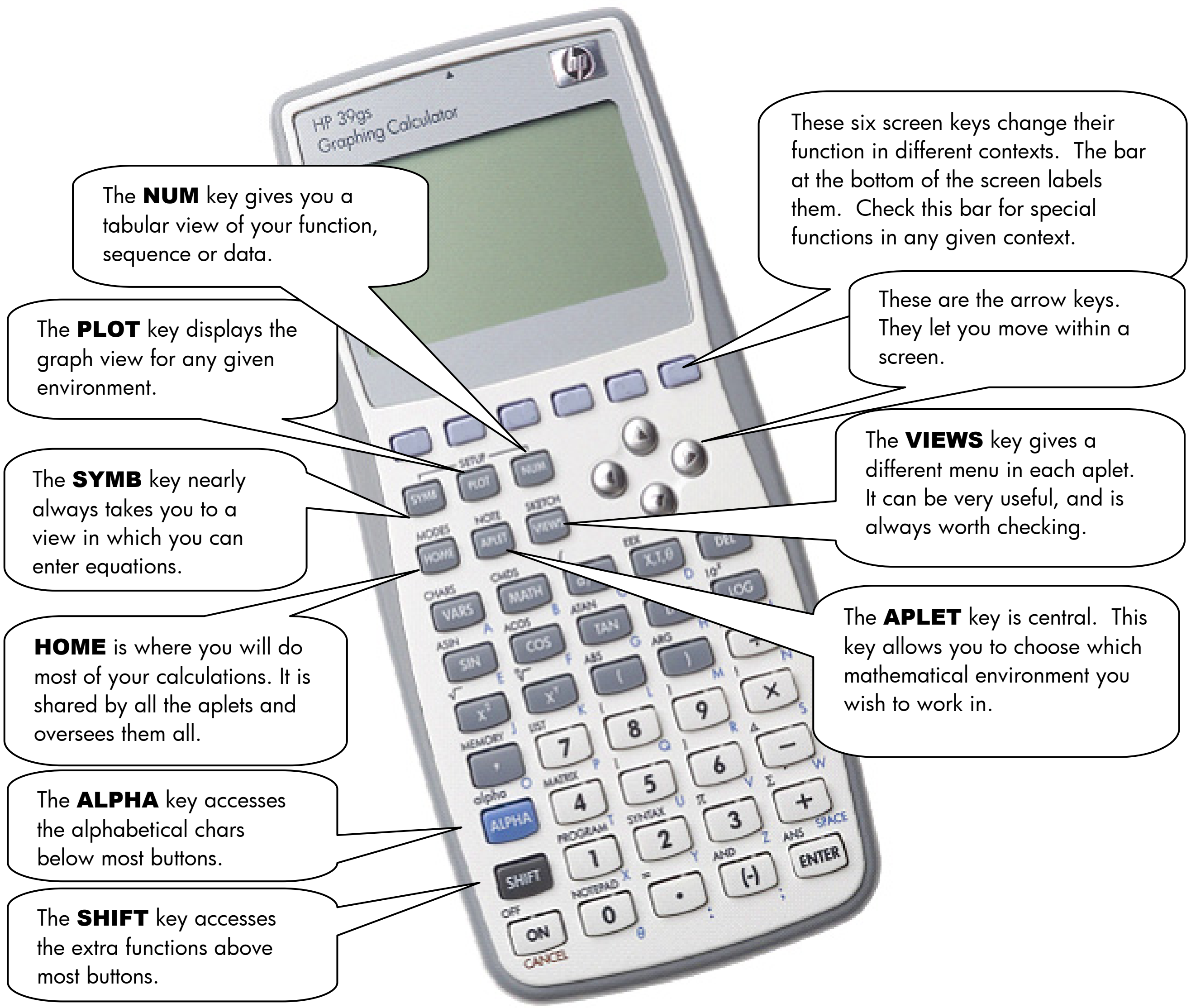
The majority of readers of this manual may only have used a Scientific calculator before so explanations are as complete as possible. However it is not the purpose of this book to teach mathematics and knowledge will largely be assumed. Those already familiar with another brand or type of calculator may find a quick skim sufficient, concentrating perhaps on the 'Expert User' chapters.

This book provides a supplement to the official manual and, more importantly, expert tips to make your work smother and more confident. It has been designed to cover the full use of the hp 39gs and hp 40gs calculators. This means explanations which will be useful to anyone from a student who is just beginning to use algebra seriously, to one who is coming to grips with advanced calculus, and also to a teacher who is already familiar with some other brand of graphic calculator.

The impact graphical calculators are having on the topics taught and even more, the way they are taught is proving to be profound. The inventiveness and flexibility of teachers of mathematics is being stretched to the limit as we gradually change the face of teaching in the light of these machines. For those concerned with the impact of a graphical calculator on the 'fundamentals' of mathematics, it should be recalled that the same fears were held for scientific calculators when they were introduced to schools. History has shown that these fears were generally groundless. Students are learning topics in high school that their parents did not cover until university years. In particular, the scientific calculator proved to be a great boon to students of middle to lower ability in mathematics, relieving them of the burden of tedious calculations and allowing them to concentrate on the concepts. It is my opinion, as a practicing mathematics teacher of some 25 years, that this is also the case with graphical calculators.

Getting Started

Let's begin by looking at the fundamentals - the layout of the keyboard and the positions of the important keys used frequently. The sketch below shows most of the important keys. As can be seen on the previous pages, the keyboards for the hp 39gs and hp 40gs are exactly the same except for the different color schemes. These keys are the ones which control the operation of the calculator – most others are simply used to do calculations once the important keys have set up the environment to do it in.



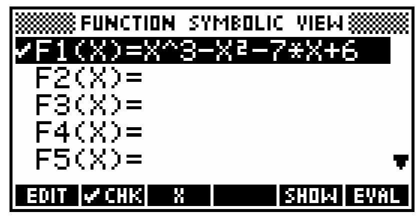
Examples of the effects of each of these keys and many more are shown on the pages that follow.

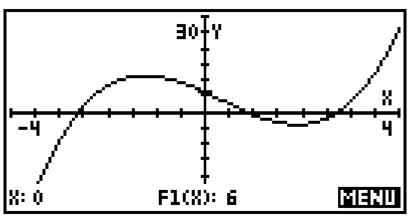
SOME KEYBOARD EXAMPLES

Shown below are snapshots of some typical screens (called "views") which you might see when you press the keys shown on the previous page. *Exactly what you see depends on which aplet is active at the time.* See page 14 for an introduction to aplets.

The Function aplet is used below to illustrate this. The normal use of the Function aplet is to graph and analyze Cartesian functions. Notice at the bottom of the various screens how the meanings of the row of unlabelled screen keys change in different views.

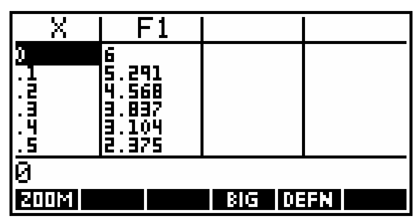
The **SYMB** key - in this case it is set to graph the function $f(x) = x^3 - x^2 - 7x + 6$.

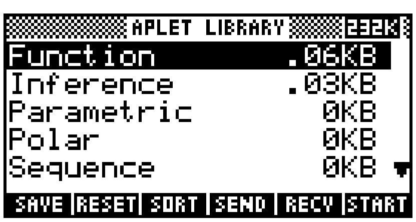




The **PLOT** key - used to graph the function. (The **PLOT SETUP** view sets the axes.)

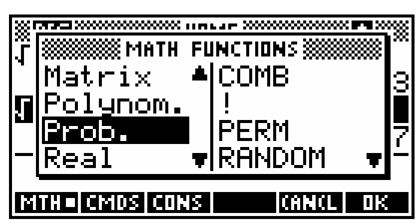
The **NUM** key showing a tabular view of the function. (The **NUM SETUP** view sets table parameters.)





The **APLET** key is used to choose which aplet is active. There are 12 aplets provided with the calculator and more can be downloaded from the internet.

The **MATH** key gives access to more than a hundred extra functions, grouped into categories. The view shown right is part of the Probability section.



KEYS & NOTATION CONVENTIONS

There are a number of types of keys/buttons that are used on the hp 39gs and hp 40gs.

The basic keys are those that you see on any calculator including scientific ones, such as the numeric operators and the trig keys. Most of these keys have two or more functions, with the second function accessed via the **SHIFT** key and the alphabetic character accessed via the **ALPHA** key.

Take for example the **COS** key shown left. If you just press the key, you get the **COS** function. However above left of the key and below right you will see two additional meanings assigned via the **SHIFT** and **ALPHA** keys.

In this book, all references to buttons, whether they need the **SHIFT** key or not, are written in this typeface: **KEY**.

The SHIFT and ALPHA keys

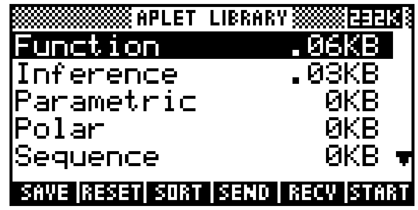
The **SHIFT** key gives you the second function for each key. In the case of the **COS** key, the second function is **ACOS**, sometimes referred to as arc-cos or $\cos^{-1}$ or inverse cos. Most keys have these second functions that are obtained via the **SHIFT** key.

Note: When I want you to use one of these keys that needs to have the **SHIFT** key pressed first I usually won't say so. It seems to me that you're intelligent enough to work out for yourself when the **SHIFT** key needs to be pressed.

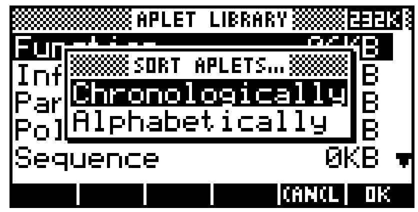
The ALPHA key

The next modifier key is the **ALPHA** key. This is used to type alphabetic characters, and these appear below and right of most keys. Pressing **SHIFT** before the **ALPHA** key will give lower case alphabetic characters. In many views the calculator can be locked in **ALPHA** mode using one of the screen keys. You can also simply hold down the **ALPHA** key.

The Screen keys



A special type of key unique to the hp 39gs, hp 40gs and family is the row of blank keys directly under the screen. These keys change their function depending on what you are doing at the time. The easiest way to see this is to press the **APLET** key. As you can see right, the functions are listed at the bottom of the screen. All you have to is to press the key under the screen definition you want to use. These buttons are normally referred to as SK1 to SK6, where SK is "Screen Key".

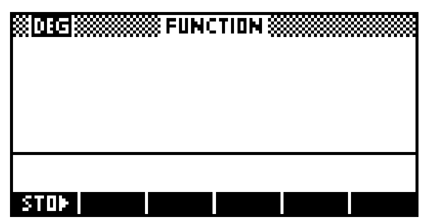


All references to keys of this type are shown as images of the label. For example, if I want you to press the key under the SORT label it would be written as **SORT**. Do it now and you'll see the screen shown on the right. Notice that the keys have now changed function. Press the one under **CANCL** to return to the previous view.

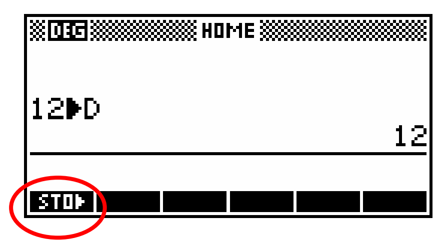
Sometimes pressing a key pops up a menu on the screen as you just saw. You use the up/down arrow keys to move the highlight through the menu and make choices by pressing the **ENTER** key. Choices that are listed in a menu will usually be written using italics. As an example, I might say to press **SORT** and choose *Chronologically*. The manual you are given with your calculator uses a different convention.

As mentioned before, the third way a key can be used is to get letters of the alphabet. This is not so that you can write letters to your friends (although you can do that with the Notepad) but so that you can use variables like X and Y or A and B. The key above the **SHIFT** key labeled **ALPHA** is used to type in letters of the alphabet. Lower case letters are obtained by pressing the **SHIFT** key before the **ALPHA** key. If you want to type in more than just a single letter, hold down the **ALPHA** key. Unfortunately, this doesn't work for lowercase.

Try this…



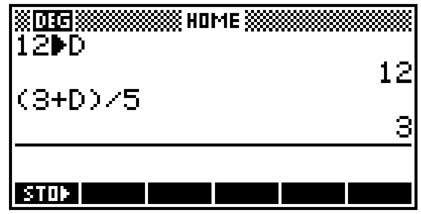
If you haven't already, **CANCL** out of the menu from the previous screen. Press the **HOME** key to see the screen on the right. Yours may not be blank like mine but that doesn't matter.



Press **1 2** and then press the screen key labeled **STO▶**, circled on the image. Now press the **ALPHA** key and then the alphabetic **D** key (on the **XT**θ key). Finally, press the **ENTER** key. Your screen should look like the one on the right. You have now stored the value **12** into memory **D**. Each alpha key can be used as a memory.
Note that memories **X, T** and θ are regularly overwritten by the normal operation of the calculator and should be used with caution.

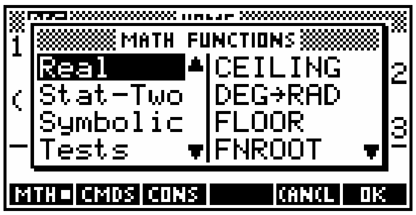
You can also use these memories in calculations. Type in the following, not forgetting the **ALPHA** key before the **D**….

(3+D)/5 ENTER



The calculator will use the value of **12** stored earlier in **D** to evaluate the expression (see image). In case you haven't worked it out for yourself, the / symbol comes from the divide key and the * symbol from the multiply key .

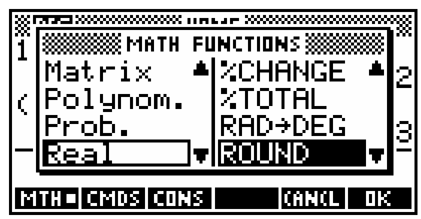
More information on memories and detailed information on the **HOME** view in general is given on pages 39.

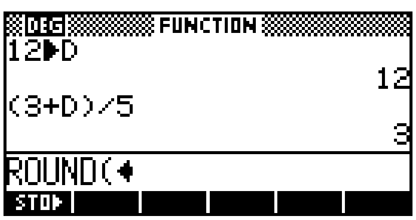


The calculator also comes with an immense number of mathematical functions. They can all be obtained via menus through the **MATH** key or from the keyboard. Try pressing the **MATH** key now and you should find your screen looks like the screen shot left.

The **MATH** menu is covered in detail on pages 165 but we will have a brief look now.

The left side of the menu lists the categories of functions. As you use the up/down arrows to scroll through the topics, you'll see the actual list on the right change. Move down through the menu until you reach *Prob.* (short for Probability) and then one step more and you'll find yourself back at *Real*. Now press the right arrow key and your highlight will move into the right hand menu (see above). Move the highlight down through this menu until you reach *Round*. Press **ENTER**.

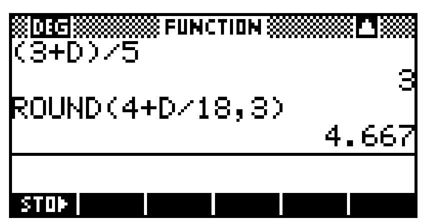




You should now be back **HOME**, with the function **ROUND(** entered in the display as shown right. You can also achieve the same effect by using **ALPHA** to type in the word letter by letter. Many people prefer to do it that way.

Now type in: **4+D/18,3)** and press **ENTER**

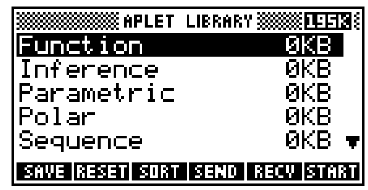
As you can see, the effect was to round off the answer of 4·666666.. to 3 decimal places. Entering **ROUND(4+D/18,-3)** would have rounded to 3 significant figures instead.



There are shortcuts for obtaining things from the **MATH** menu that are covered later (see page 41).

6

EVERYTHING REVOLVES AROUND APLETS!



A built in set of aplets are provided in the **APLET** view on the hp 39gs and hp 40gs. This effectively mean that it is not just one calculator but a dozen (or more), changing capabilities according to which aplet is chosen.

The best way to think of these aplets is as “environments” or “rooms” within which you can work. Although these environments may seem dissimilar at first, they all have things in common, such as that the **PLOT** key produces graphs, that the **SYMB** key puts you into a screen used to enter equations and rules, and that the **NUM** key displays the information in tabular form.

There are twelve standard aplets available via the **APLET** key. More can be created by you or obtained via the Internet (see pages 255 & 245). These aplets are:

The Finance aplet (see page 155)

Performs calculations involving time/value of money.

The Function aplet (see page 46)

Provides f(x) style graphs, calculus functions etc. It will not only graph but find intercepts, intersections, areas and turning points.

The Inference aplet (see page 141)

Allows the investigation of inferential statistics via hypothesis testing and confidence intervals.

The Linear Solver aplet (see page 150)

This aplet is used to solve simultaneous linear equations in two or three unknowns.

The Parametric aplet (see page 92)

Handles x(t), y(t) style graphs. Can also be used to help with vector motion.

The Polar aplet (see page 98)

Handles r(θ) style graphs. Quite apart from their mathematical use, they produce some really lovely patterns!

The Quadratic Explorer aplet (see page 159)

This is a teaching aplet, allowing the student to investigate the properties of quadratic graphs using an interactive format which allows the connection between the coefficients and the graph to be easily seen.

The Sequence aplet (see page 99)

Handles sequences such as $T_n = 2T_{n-1} + 3;\ T_1 = 2$ or $T_n = 2^{n-1}$. Allows you to explore recursive and non-recursive sequences.

The Solve aplet (see page 105)

Solves equations for you. Given an equation such as $A = 2\pi r\left(r + h\right)$ it will solve for any variable if you tell it the values of the others.

The Statistics aplet (see page 114 & 123)

Handles descriptive statistics. Data entry is easy, as is editing. It analyzes univariate and bivariate data, drawing scatter graphs, histograms and box & whisker graphs and finding lines of best fit, linear and non-linear.

The Triangle Solve aplet (see page 152)

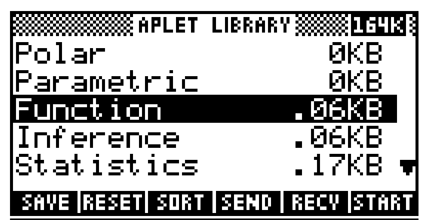
This aplet solves for sides and angles in triangles.

The Trig Explorer aplet (see page 162)

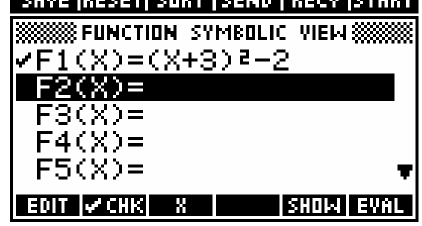
This is a teaching aplet, allowing the student to investigate the properties of sine and cosine graphs in the same interactive fashion as the Quadratic Explorer.

The Function aplet is probably the easiest to understand and also the one you will use most often, so we will have a very quick look at the commonly seen views of this aplet.

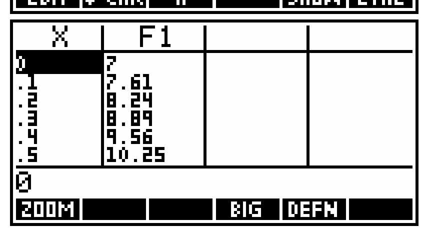
The **APLET** key is used to list all the aplets and to START, RESET or SAVE them.



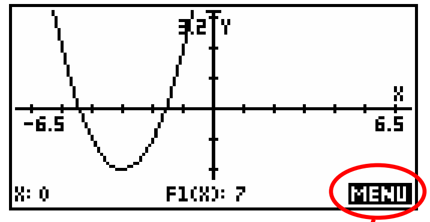
The **SYMB** view is used to enter equations….
It can store up to ten functions. If you SAVE additional copies of the aplet then any number of functions can be used.



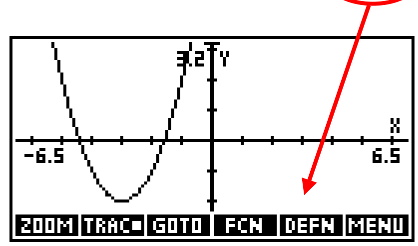
The **NUM** view shows the function in table form…



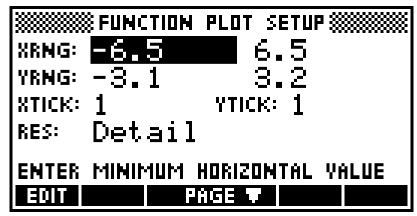
The **PLOT** view is used to display the function as a graph…



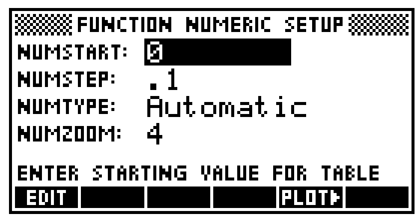
At the bottom of the **PLOT** view is the MENU key. This gives access to a number of other useful tools allowing further analysis of the function. Some of these tools have their own sub-menus.



The **PLOT SETUP** view controls the settings for the **PLOT** view…



The **NUM SETUP** view controls the settings for the **NUM** table view…



Although these views are superficially different in other aplets, the basic idea is usually similar.

Having said that aplets are best thought of as “working environments”, it is equally true that aplets are essentially programs, with the standard ones simply being built into the calculator. This is a programmable calculator, having its own programming language and able to perform quite sophisticated tasks.

Unless you particularly want to learn about the programming language, there is no reason why you should worry about it. The standard aplets will cover all of your normal requirements in mathematics.

However one of the great strengths of the hp 39gs and hp 40gs is their ability to “download” additional aplets from other calculators and from the Internet. See page 245.

A mini-USB cable (see page 237) and software were provided with your hp 39gs and hp 40gs which you can use to connect your PC to your calculator via the USB port and then download aplets from the computer to the calculator or to save your work to the computer. If you have an earlier model such as the hp 38g, hp 39g or hp 40g then you need to buy the cable separately from an HP reseller and download the software from Hewlett-Packard's website. More information on this can be found on pages 245 - 274.

Calculator Tip
If you search on the web using the key words "39g" or "40g" and you will find a variety of sites which contain information and aplets. A large variety of programs and aplets were written for the older hp 39g & hp 40g and these will generally run with no problems on the hp 39gs & hp 40gs. The author's site (www.hphomeview.com) has the largest collection and has extensive help pages as well.

Once an aplet is transferred onto any calculator from the PC, transferring it to another takes only seconds using the built in infra-red link at the top of each calculator on the hp 39gs or using the mini-serial cable on the hp 40gs. The infra-red link is exactly like the remote control of a TV or VCR, and allows two calculators to talk to each other. In the interests of security in examinations the distance over which the infra-red link can communicate is limited to about 8 - 10cm (about 3 - 4 inches). See page 237 for details on this process. The hp 40gs does not have infra-red because some of the markets for which it was produced did not wish students to have this capability.

Aplets are available to do many mathematical tasks such as statistical simulations, time series analysis as well as many tasks called for in calculus, physics and chemistry. There are a number of sites which offer aplets.

The Hewlett-Packard site is found at…

http://www.hp.com/calculators
(follow the links to graphical calculators and then your model)

In addition to this you should check the site called *The HP HOME view* which can be found at…

http://www.hphomeview.com
(This site is maintained by the author and contains not only hundreds of aplets and games, but also a huge amount of detailed information on the 39gs/40gs family of calculators.)

The entire topic of aplets is discussed in more detail in the chapter entitled "Copying & creating aplets" on page 226.

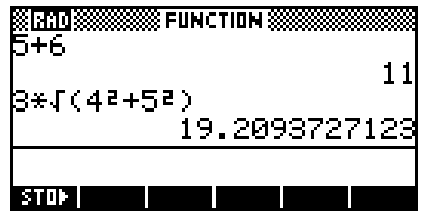
Calculator Tip
The aplets for an hp 39/40 family are interchangeable but those of an hp 38g are not. If you load an aplet from an hp 38g onto an newer model then the download may appear to be successful but the calculator will "crash" when the aplet is run. No permanent damage will be done but it may cause a reset with loss of user information. Programs from the hp 48 and 49 family are not compatible with those of the 39/40 family (nor vice versa).

THE HOME VIEW

In addition to the aplets, there is also the **HOME** view, which can best be thought of as a scratch pad for all the others. This is accessed via the **HOME** key and is the view in which you will do your routine calculations such as working out 5% of $85, or finding √35. The **HOME** view is the view that you will most often use, so we will explore that view first.

What is the HOME view?

This is the **HOME** base for the calculator. All other aplets can be accessed from it and can affect it to varying degrees. All mathematical functions are available in this view. You should learn to use this view as efficiently as possible, since a great deal of work will be done here.



We will explore the **HOME** view in the following order:

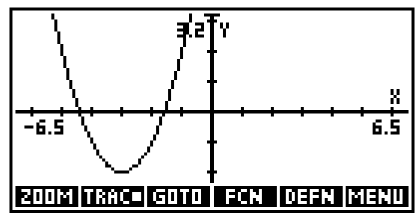
- Exploring the Keyboard
- Angle and numeric settings
- Memory management
- Fractions on the hp 39gs & hp 40gs
- The **HOME** History
- Storing and retrieving memories
- Referring to other aplets from the **HOME** view
- An introduction to the **MATH** menu
- Resetting the calculator

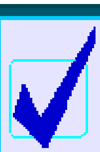
Exploring the keyboard

The first step in efficient use of the calculator is to familiarize yourself with the mathematical functions available on the keyboard. If we examine them row by row, you will see that they tend to fall into two categories - those which are specific to the use of aplets, and those which are commonly used in mathematical calculations.

The first row of blank keys are context defined. The reason they have no label is that their meaning is redefined in different situations - they are the 'screen keys'. The current meaning of each key is listed in the row of boxes at the bottom of the screen.

A common abbreviation used for these keys is **SK1** or **SK2** etc (for "screen key 1"). In the **PLOT** view shown right, some of the screen keys are labeled, such as the MENU key. When you press this MENU key the row of screen keys labels in the **PLOT** view appear or disappear. To see another view where all the keys are in use, change to the **APLET** view.

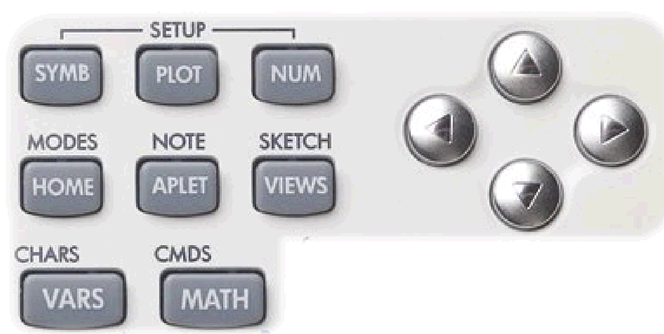




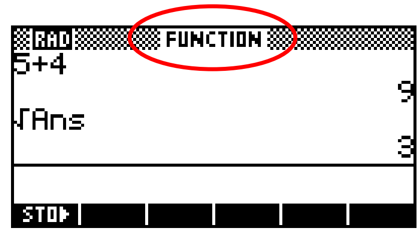
Calculator Tip
Develop the habit of checking the screen to see if any of those keys have been given meanings. In many views, the screen keys have been set up with useful shortcuts and functions.

The next two rows of keys and part of the third are mainly aplet related, so we'll deal with them as a group.

The arrow keys on the right are used in most views, usually to move the cursor (a small cross) or the highlight around on the screen.



The **APLET** key is used to choose between the various different aplets available. Everything in the calculator revolves around aplets, which you can think of either as miniature programs or as environments within which you can work.



The hp 39gs and hp 40gs come with twelve standard aplets - Finance, Function, Inference, Parametric, Polar, Linear Solver, Triangle Solver, Quadratic Explorer, Sequence, Solve, Statistics and Trig Explorer. Which one you want to work with is chosen via the **APLET** key. See page 14 for more details on this.

Calculator Tip

The name of the active aplet is shown at the top of the screen, as above. It is important to bear this in mind because the angle and numeric settings are tied to the active aplet. Changing aplets may therefore cause these settings to change in **HOME** too. See page 28.

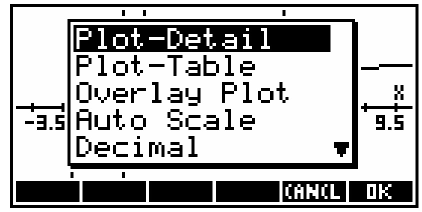
In addition to the standard twelve, covered in great detail in the chapters following, many more aplets are available from the Internet written by other programmers. Once these are downloaded into your calculator they can also be accessed via the **APLET** key. For more detail on this type of aplet, see the brief summary later in this section, and the chapter entitled "Programming the hp 39gs & hp 40gs" on page 255.

When working mathematically there are three ways that we view functions:

- symbolically, as an equation;
- numerically, as a table;
- graphically, as a graph in various formats.

The **SYMB**, **PLOT** and **NUM** keys are intended to reflect this common methodology. In most aplets the **PLOT** view shows the graph, the **SYMB** view shows the equations and the **NUM** view shows the equations in tabular (numeric) format.

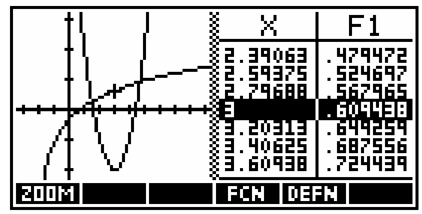
The **VIEWS** key pops up a menu from which you can choose various options. Part of the **VIEWS** menu for the Function aplet is shown right. See page 85 for more detailed information. A summary only is given below.



Essentially the **VIEWS** menu is provided for two purposes…

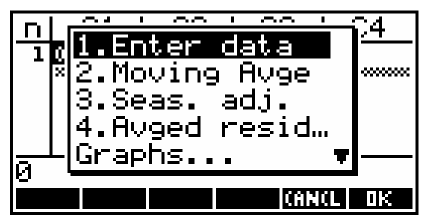
Firstly, within the standard aplets (Function, Sequence, Solve etc.) it provides a list of special views available to enhance the **PLOT** view.

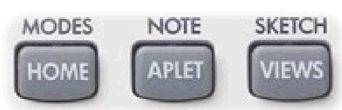
For example the standard **PLOT** screen provides a standard graph covering the whole screen, but the **VIEWS** menu lets you use a split screen such as shown right. Information on the **VIEWS** menu is given in the chapter dealing with the Function aplet.



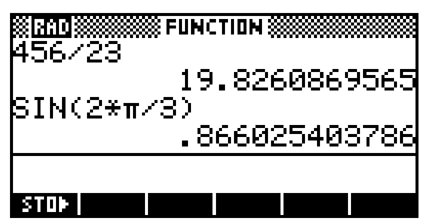
In its second role, the **VIEWS** key also has a critical purpose when using aplets which have been downloaded from the Internet. When a programmed aplet is created for the hp 39gs or hp 40gs, a menu is provided by the programmer to let you control and use it. During the programming this menu is tied to the **VIEWS** key, replacing the menu normally found on the key.

For example, the snapshot shown right is of a **VIEWS** menu taken from an aplet designed to analyze and graph Time Series data.

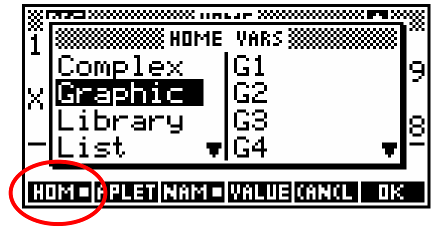




The next important key is the **HOME** key. It allows you to change into the **HOME** view from wherever you are. Above it is the **MODES** key, accessed by pressing **SHIFT** first. More detailed information on these two views follows later.

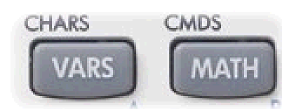


The **VARS** key is used, mainly by programmers, as a compact way to access all the different variables stored by the calculator including aplet environment variables.

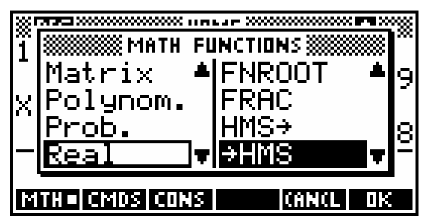


Shown right are two views of the **VARS** screen, the first from the **HOME** list showing the graphic variables (memories) **G1**, **G2**.... and the next from the **APLET** list showing some of the variables in the set controlling **PLOT**.

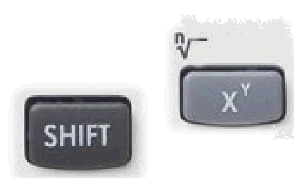
The **VARS** key is not generally used much, and you may not have followed this explanation. This is not important as it is a key that is very rarely used by the average user. A few uses for the average user are detailed in the Function aplet's "Expert User" section on page 62.



The **MATH** key next to **VARS** is far more important and provides access to a huge library of mathematical functions. The more common functions have keys of their own, but there is a limit to the number of keys that one can put on a calculator before it takes too long to find the key required. Hence the **MATH** key.



The **MATH** menu lists all those functions that would not fit onto the keyboard plus some which also appear on the keyboard. Shown in the screen snapshot right is a small selection of the total list. For a listing of almost all the functions, with examples of their use, see the chapter entitled "The MATH Menus" on page 165.



As is usual with all calculators, most of the keys have another function above the key. The hp 39gs and hp 40gs get twice the action from each key by having this second function.

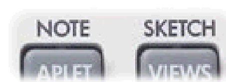
The second function is accessed via the **SHIFT** key on the left side of the calculator. Although this book will sometimes tell you explicitly to press this key, in most cases it will be assumed that you are intelligent enough to work out for yourself when it is necessary to press it.

The **ALPHA** key gives access to the alphabetical characters, shown below and right of most keys. Pressing **SHIFT ALPHA** gives lower case.

Calculator Tip
If you press and hold down the **ALPHA** key you can 'lock' alpha mode, although this doesn't work for lower case. Many people use this to type in functions by hand rather than going through the **MATH** menu. Some views, such as the Notepad, also offer a screen key function that lets you lock either upper or lower case alpha mode.

The **SETUP** views, above **PLOT**, **SYMB** and **NUM**, are used to customize their respective views. For example, the **PLOT SETUP** screen controls things like axes, labels etc. Their use changes in different aplets, so for more information see the explanations in the chapters dealing with the various aplets, particularly in the Function aplet on page 50.

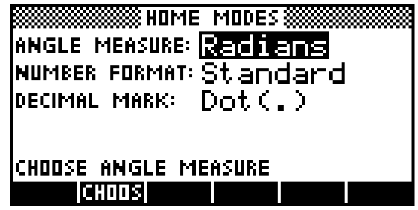
The **SYMB SETUP** key is only used in one place, which is to choose the data model for bivariate statistics in the Statistics aplet. It is not available in the other aplets and trying to access it will result only in a quick flash of an exclamation mark on the screen to say "You've done something wrong!".

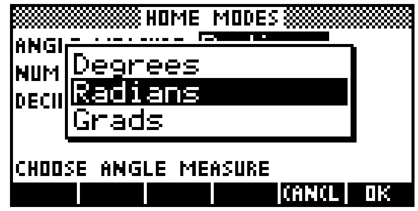


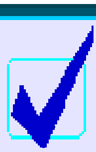
Information on the use of the **SKETCH** and **NOTE** views (located above the **APLET** and **VIEWS** keys) can be found in the chapters "Working with Sketches" and "Working with Notes & the Notepad" on pages 222 & 217.

The main use for the **SKETCH** and **NOTE** views is in aplets downloaded from the Internet. Instructions for using the aplet are sometimes included with the aplet in note form, and sometimes as an accompanying sketch.

The **MODES** view (see right) controls the numeric format used in displaying numbers and angles in aplets. At the bottom of the screen you will see that one of the screen keys has been given the function **CHOOS**. Pressing this key pops up a menu of choices from which you can select the option which suits you. The default angle setting is radians.





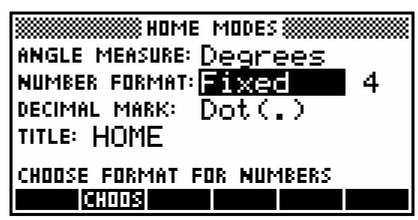


Calculator Tip

If you don't want to use the menu then, rather than pressing **CHOOS**, highlight the field and then press the '**+**' key repeatedly. This will cycle through the choices without popping up a menu. This can be much faster if the menu has only a few choices.

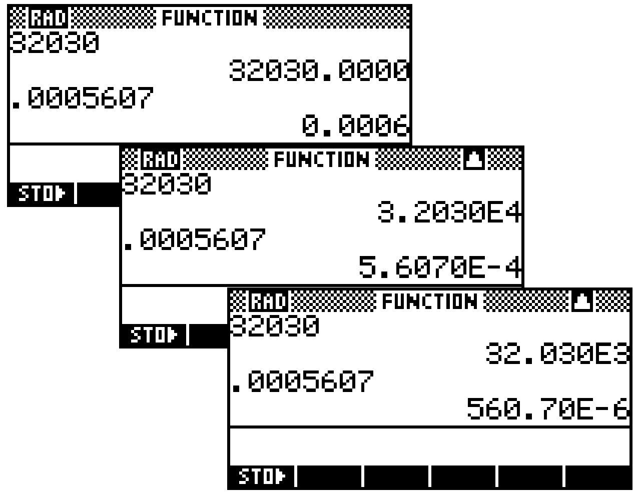
The choices for '**Number format**' are shown on the right. **Standard** is probably the best choice in most cases, although it can be a little annoying to constantly have 12 significant figures displayed. In **Standard** mode, very large and very small numbers are displayed in scientific notation.

The **Fixed**, **Scientific** and **Engineering** formats all require you to specify how many decimal places to display. The screenshot right shows **Fixed 4**, which rounds everything off to 4 decimal places. Of course, you can change the **4** to any other number you want.



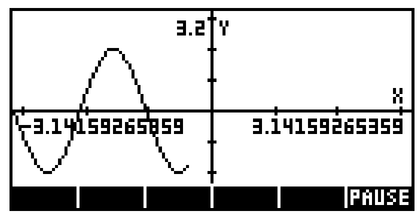
A setting of **Scientific** ensures that any results are displayed in scientific notation. Of course, the calculator's idea of scientific notation may not be the same as yours. Since the calculator has no way of displaying powers as superscripts, a result of $3 \cdot 203 \times 10^4$ has to be displayed as **3.203E4**. The alternative of **Engineering** is very similar to **Scientific**, except that powers are displayed as multiples of 3 (ie. $10^3, 10^6$). This is done to allow easy conversion in the metric system, which also works in multiples of 1000.

The screens right show the same two numbers displayed as in turn as; **Fixed 4**, **Scientific 4** and **Engineering 4**.



Calculator Tip
If you have **Labels** turned on when you **ZOOM** in (or out) on a graph or choose a Trig scale then you may end up with axes whose numeric labels are horrible decimals (see below right).

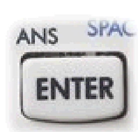
The setting of **Fraction** can be quite deceptive to use and is discussed in more detail on page 33.



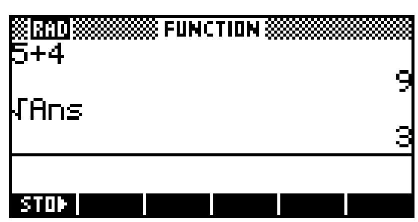
The next alternative in the **MODES** view of **Decimal Mark** controls the character which is used as a decimal point. In some countries a comma is used instead of a decimal point.

If you opt to use a comma rather than a full stop then any places where a comma would normally be used (such as in lists) will swap to using a full stop. Any functions which might normally have terms separated by a comma will use a full stop instead. For example, **ROUND(3.456,2)** will become **ROUND(3,456.2)**.

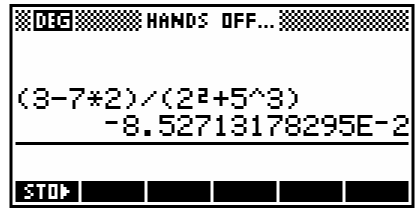
Moving back to our tour of the keyboard, the next key is the **ENTER** key. This is used as an all purpose "I've finished - do your thing!" signal to the calculator.



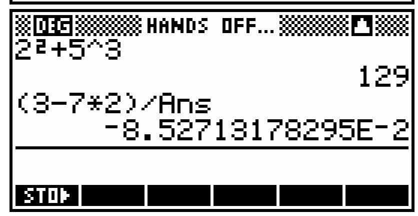
In situations where you would normally press the '**=**' key on most calculators, press the **ENTER** key instead. Above the **ENTER** key is the **ANS** key. This can be used to retrieve the final value of the last calculation done. An example is shown right.



If you are not confident about using brackets, then the **ANS** key can be quite useful.

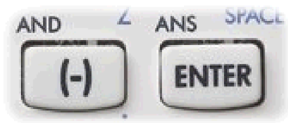


For example, you could calculate the value of $\frac{3-7\times 2}{2^2+5^3}$ by using brackets…



…. or you could use the **ANS** key.

A better alternative to using the **ANS** key is to use the History facility and the **COPY** function. This is discussed on page 37.

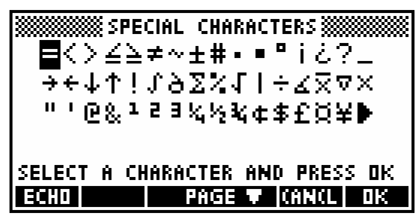


Another important key is the **(-)** key shown left. It is important to realize that the hp 39gs and hp 40gs do not treat a negative as being the same as a subtract.

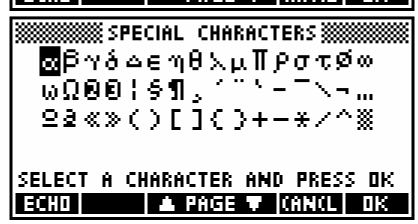
If you want to calculate the value of (say) $-2-(-9)$ then you must use the **(-)** key before the 2 and the 9 rather than the subtract key. If you press the subtract key twice, entering 'subtract, subtract 9' instead of 'subtract (-) 9' you will receive an error message of "Invalid syntax", meaning it does not make mathematical sense to have two subtract signs rather than a subtract of a negative.

Similarly, if you press 'subtract 2' as the first keys in the above calculation then the calculator will display **Ans - 2**. The reason for this is that a subtract cannot start an expression in mathematics, while a negative sign can. Since the subtract can't come first, the calculator decides that you must have intended to subtract from the previous answer. Hence the sudden appearance of an **Ans**. This occurs at other times too. A common error by new users is to enter a value into the **PLOT SETUP** view using subtract instead of negative. This will usually have unexpected results.

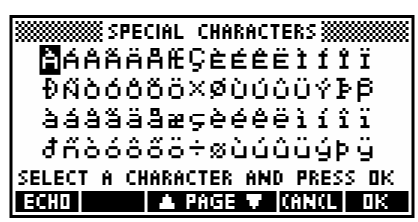
The next important key is the **CHARS** key (above **VARS**). It accesses a view containing the characters that are required occasionally but not often enough to bother putting on the keyboard.



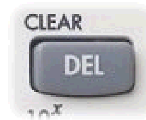
Pressing **SHIFT CHARS** will pop up the screens shown right. One of the screen keys is set to be a 'Page Down' key **PAGE ▼**, and will give access to two more pages of characters as shown.



These special characters are obtained by pressing the screen key labeled **ECHO**. You can press **ECHO** as many times as you need to in order to obtain multiple characters. When you have as many as you need, press the **OK** key. If you only require one character then **ECHO** is not required – just press **OK**.



Another important key is the **DEL** key at the top right of the keyboard. This serves as a backspace key when typing in formulas or calculations, erasing the last character typed. If you have used the left/right arrow keys to move around within a line of typing, then the **DEL** key will delete the character at the cursor position.



The **CLEAR** key above **DEL** can be thought of as a kind of 'super delete' key. For example, if pressing **DEL** would erase one function only in the **SYMB** view then **CLEAR** will erase the whole set.

Calculator Tip

Another use for the **DEL** key is to restore factory settings. For example, if you move back into the **MODES** screen and change to Degree mode, then pressing the **DEL** key will restore the default of Radians. Pressing **CLEAR** in the **MODES** view would restore factory settings to all the entries. This is particularly useful in the **PLOT SETUP** view.

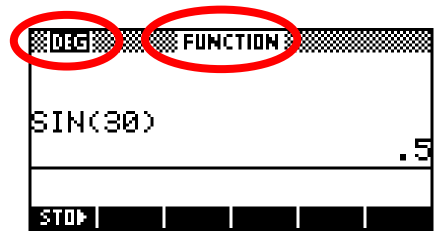
The remaining keys of **LIST**, **MATRIX**, **MEMORY**, **NOTEPAD** and **PROGRAM** have special chapters of their own.

Angle and Numeric settings

It is critical to your efficient use of the hp 39gs and hp 40gs that you understand how the angle and numeric settings work. For those few who may be upgrading from the original hp 38g released in the mid '90s this is particularly important, since the behavior is significantly different.

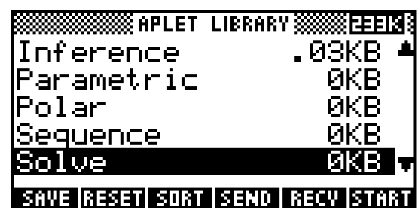
On the hp 39gs and hp 40gs, when you set the angle measure or the numeric format in the **MODES** view, it applies both to the aplet and to the **HOME** view. However, this setting applies *only* to the currently active aplet (the one highlighted in the **APLET** view).

This means that if you change active aplet then these settings may change also, not only for the aplet but in the **HOME** *view too.*

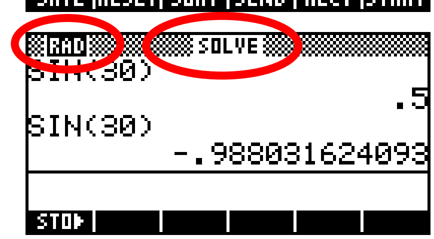


For example, suppose you have been performing trig calculations in the **HOME** view with the Function aplet being currently active, and have set the angle measure to **DEG**.

If you were to now change to the Solve aplet in order to solve an equation then the settings would revert to those of the Solve aplet; probably be radians unless you had also changed those as well.

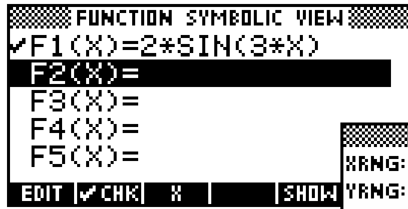


Radian measure is the default for all aplets and thus also for **HOME** unless you change it. Performing exactly the same calculation in **HOME** would now give a different result, as shown right.

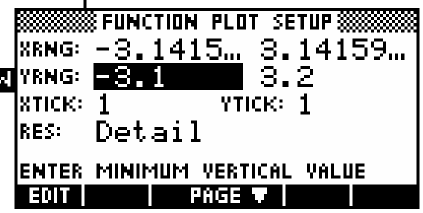


Although this may seem to be a strange way of doing things, there is actually a very good reason for it. On the original first model, the hp 38g, the settings for the **HOME** view were independent from those of the aplets and it caused a number of users to have difficulties as you'll see on the next page.

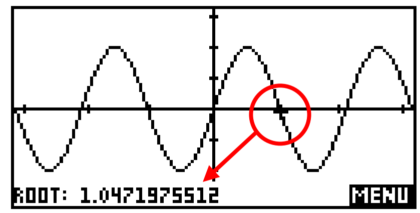
Suppose we define a trig function in the Function aplet as shown.



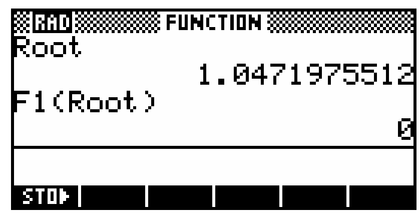
The default setting for the Function aplet is radians, so if we set the axes to extend from $-\pi$ to π, the graph would look as shown right.



In the **PLOT** view shown, the first positive root has been found (see page 57) as x=1.0471…

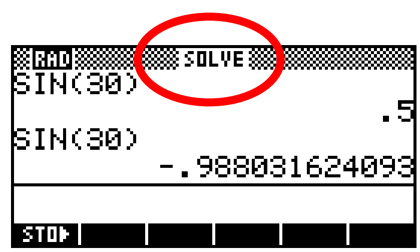


On the hp 39gs and hp 40g, if we now change to the **HOME** view, retrieve the root and perform the calculation shown right, we expect that the answer should be zero, as indeed it is.

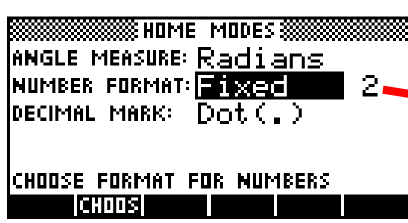


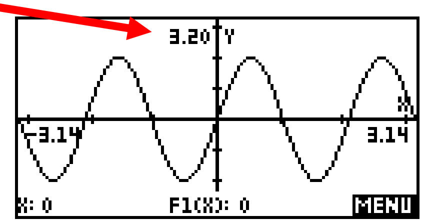
However, this is only the case because the angle measures of **HOME** and the Function aplet agree. The problem was that on the original hp 38g the default setting for the Function aplet was radians, while **HOME** had a default setting of degrees and its setting was independent of those of the aplet. This meant that a calculation such as the one above would give incorrect results, and caused considerable confusion to some students. It even resulted in users returning their hp 38g to dealers as being 'faulty'! Hence the change, which was first made in the hp 39g and hp 40g.

The only drawback of synchronizing the settings of the **HOME** and aplet is that you might change aplets and forget that it may also change the **HOME** view settings. For this reason, the name of the active aplet is shown at the top of the **HOME** view as a reminder.



On the hp 39gs and hp 40gs you can see that if we turn **Labels** on and then **PLOT**, the numeric mode also affects the axis labels.





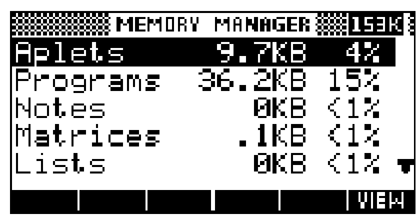
The default behavior in the **PLOT** view is to not display axis labels due to the way that they often interfere with the clear display of the graph.

Settings made in the **MODES** view also apply to the appearance of equations and results displayed using the **SHOW** command, covered on page 38.

Calculator Tip
Under the system used on the hp 39gs and hp 40gs, if you want to work in degrees then you will need to choose that setting in the **MODES** view and possibly set it again if you change to another aplet. Some people choose to go through and change the setting on all the aplets at once so that they don't have to remember that it might change. However, if you **RESET** an aplet or reset the calculator then the default setting will return.

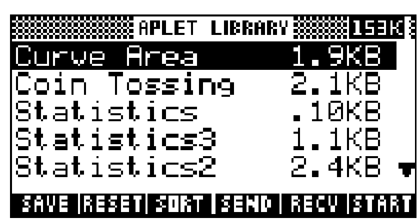
Memory Management

One of the major complaints about the original hp 38g was its memory - mainly the lack of it at only 23Kb, but also the inability to easily control or manage it. This problem has been addressed on the hp 39gs and hp 40gs in two ways. Both have a very ample amount, just short of 200Kb, and there are very few users who will come close to filling this. Depending on size, there is enough room for at least 40 aplets, 40 pages of notes, or nearly 10,000 data points although not, of course, all at once.



In addition to all this memory, the hp 39gs and hp 40gs supply an easy way to control it through the **MEMORY MANAGER** view. If you press the **MEMORY** key you will see the view shown right. Scrolling through it will show you exactly how the available memory is currently being used. The remaining memory, in Kb, is shown at the top right of the screen. This view gives an overview of the memory. For detailed management the **VIEW** key is provided.

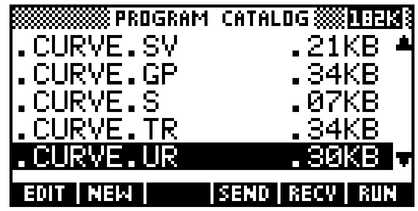
Pressing **VIEW** on any entry will take you a relevent screen in which you can delete entries no longer needed.

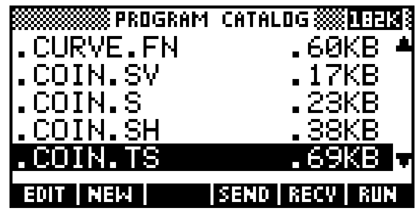


For example, with the highlight on *Aplets*, pressing **VIEW** will take you to the **APLET** view (right), where you can choose to delete or reset any aplets no longer required.

As you can see in the screen snapshot on the previous page, my calculator has a number of extra aplets. Two of them, *Statistics2* and *Statistics3* are simply copies of the normal Statistics aplet containing data that I did not want to lose. The top two aplets *Curve Area* and *Coin Tossing* are teaching aplets that I have downloaded from the internet.

If you use teaching aplets that you download from the internet via the Connectivity Kit, or which are supplied to you by your teacher via the infra-red link on an hp 39gs or the cable on an hp 40gs, then you need to bear in mind that most of them have 'helper' programs that aid them in performing their tasks.



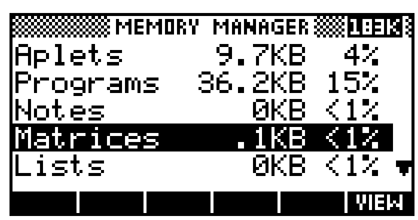


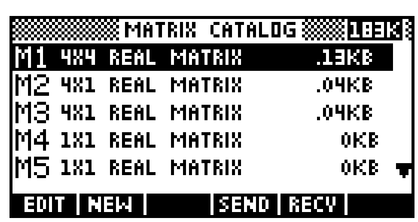
In the screens shown right you can see some of the programs which are attached to the two aplets mentioned above. The convention which most programmers follow is to name these 'helper' programs in a way that associates them clearly with the parent aplet. The memory associated with these programs is not included in that shown for the aplet in the **APLET** view but will not usually be a very large amount, as you can see in the examples shown.

Calculator Tip

The reason for this naming convention for 'helper' programs is that when you delete the parent aplet in the **APLET** view the 'helper' programs are NOT automatically deleted with it because they may be shared by other aplets. You must change to the **Program Catalog** view and delete them manually after you have finished with the teaching aplet and deleted it in the **APLET** view. If you don't do this then they will continue to take up memory on the calculator. Even on the hp 39gs and hp 40gs this is not infinite and too many left over programs will eventually cause problems.

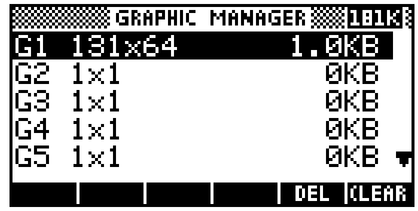
As mentioned on the earlier, pressing **VIEW** in the **MEMORY MANAGER** screen takes you to a relevent view showing greater detail. For example, the *Matrices* entry right shows 0.1 Kb in use. Pressing **VIEW** will take you to the **MATRIX CATALOG** view which shows exactly where the memory is being used and allows you to delete any or all of the matrices. Alternatively, you can enter the view in the normal way by pressing **SHIFT 4**.

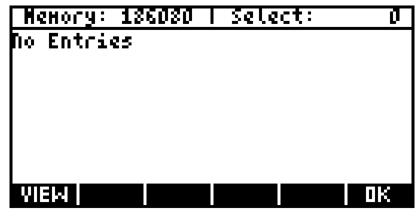




The *History* entry will take you to the **HOME** view, where pressing **SHIFT CLEAR** will clear the History.

There are two views, shown right, for which the only access is via the **MEMORY MANAGER** screen. The first of these, the **GRAPHICS MANAGER**, shows some memory in use on my calculator due to the screen captures I am performing to show you these views. Yours will probably be empty. If you have loaded an aplet from the internet then it may have used a GROB (Graphics Object) to store an image as part of its working.





The second view is the **LIBRARY MANAGER**, and this will almost certainly be empty unless you have games loaded. Generally, the only aplets which use libraries are those such as games which are written by expert programmers in machine code in order to make them run as fast as possible. These games, available on the internet, are listed in the **APLET** view along with the normal aplets and when you delete them the associated library *is* automatically deleted with them, unlike the case of the 'helper' programs.

Calculator Tip

1. Because of the amount of memory available on the hp 39gs & hp 40gs, the Memory View is not one that you will normally need to worry about unless you store a truly amazing number of Notes. It is probably of more interest to programmers.

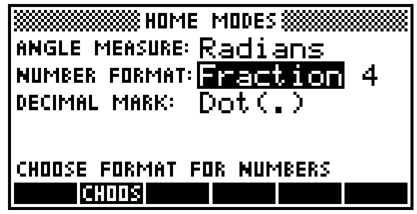
2. As discussed later, most games currently available for the hp 39gs and hp 40gs were originally written for the hp 39g. This can cause problems in two respects:

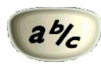
   - The hp 39g was a much slower calculator and the games may simply run too fast to be playable.

   - Some games were written using programming commands specifically aimed at the hp 39g chip. When these commands execute on an hp 39gs or hp 40gs they may cause the calculator to lock up and/or lose user memory. They should not, however, cause permanent damage.

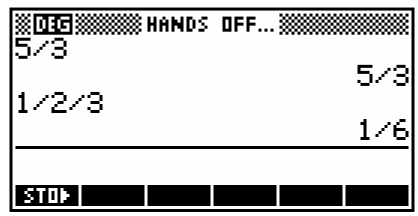
Fractions on the hp 39gs and hp 40gs

Earlier we examined the use of the **MODES** view, and the meaning of **Number Format**. We discussed the use of the settings **Fixed**, **Scientific** and **Engineering**, but left the setting of **Fraction** for later. The reason for this is that the **Fraction** setting can be a little deceptive. Begin by selecting **Fraction** in the **MODES** view, leaving the accompanying number as the default value of **4**.



Most calculators have a fraction key, often labeled , that allows you to input, for example, $1\frac{2}{3}$ as $1\neg2\neg3$ or something similar. What these calculators usually won't do is allow you to mix fractions and decimals. A calculation such as $1\frac{2}{3}+3\cdot7$ will usually give a decimal result: most calculators will not attempt to convert the $3\cdot7$ into a fraction. The reason for this is that while some decimals like 0.25 are easy to convert to a fraction, others, such as recurring ones, are not so easy. Most calculators opt for the easy option of switching to a decimal answer in any mixture of fractions and decimals. When making the hp 39gs and hp 40gs HP took a very different approach. Once you select **Fraction** mode, all numbers become fractions - including any decimals.

The first point to remember is that there is no provision for inputting mixed fractions such as $1\frac{2}{3}$. Fractions are entered using the divide key and, while the calculator is quite happy with improper fractions such as **5/3**, it correctly interprets **1/2/3** as ½ divided by 3 and gives a result of **1/6** . The solution to this is simply to enter mixed fractions as **(1+2/3)**.



Calculator Tip

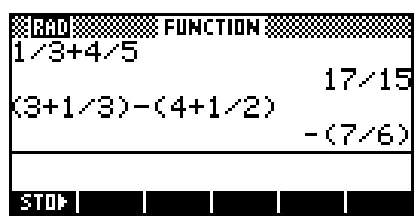
You need to be careful with brackets or else "order of operations" problems may occur, such as $1\frac{2}{3} * \frac{1}{5}$ being interpreted as **1+(2/3*1/5)** rather than as it should be: **(1+2/3)*1/5**.

When in doubt, use brackets for mixed fractions.

Some examples are… (using a setting of **Fraction 4** or higher)

$$\frac{1}{3}+\frac{4}{5}=\frac{17}{15}$$

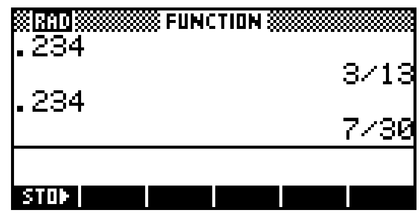
$$3\frac{1}{3}-4\frac{1}{2}=-1\frac{1}{6}$$



The second point to remember involves the method the hp 39gs and hp 40gs use when converting decimals to fractions, which is basically to generate (internally and unseen by you) a series of continued fractions which are *approximations* to the decimal entered. The final fractional approximation chosen for display is the first one found which is 'sufficiently close' to the decimal. Look up 'continued fractions' on the web or in a textbook if you don't know what these are.

The trap lies in what constitutes 'sufficiently close', and this is determined by the '**4**' in **Fraction 4**. Very roughly explained, the calculator will use the first fraction it finds in its process of approximation which matches the decimal to that number of significant digits.

For example, a setting in the **MODES** view of…



Fraction 1

changes 0.234 to $\frac{3}{13}$

which is actually 0.2307692…

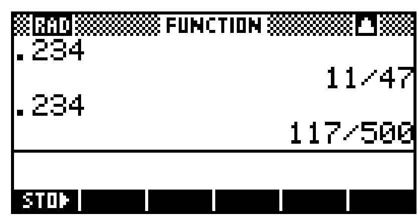
(matching to at least 1 significant figures.)

Fraction 2

changes 0.234 to $\frac{7}{30}$

which is actually 0.2333333…

(matching to at least 2 significant figures.)



Fraction 3

changes 0.234 to $\frac{11}{47}$

which is actually 0.2340425…

(matching to at least 3 significant figures.)

Fraction 4

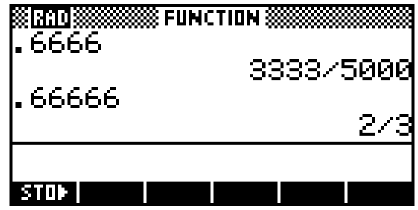
changes 0.234 to $\frac{117}{500}\left(or\ \frac{234}{1000}\right)$

which is exactly 0.234

(thus finally matching to the required 4 significant figures.)

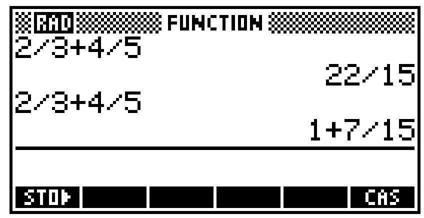
Essentially, the value of **4** in **Fraction 4** affects the degree of precision used in converting the decimal to a fraction. As was said earlier, the calculator will use the first fraction it finds in its process of approximation which matches the decimal to that number of significant digits.

The **Fraction** setting is thus far more powerful than most calculators but can require that you understand what is happening. It should also be clear now why a special fraction button was not provided: the ‘fractions’ are never actually stored or manipulated as fractions at all!



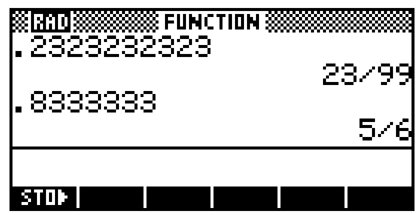
As you can see above right, a setting of **Fraction 4** produces a strange (but actually correct) result for 0.666, while adding one more 6 (to take the decimal beyond 4 d.p.) will give the desired result of **2/3.** In other words, so long as you understand the approach taken by the hp 39gs and hp 40gs it is capable of producing results which are closer to what was probably intended by the user in entering 0.66666.

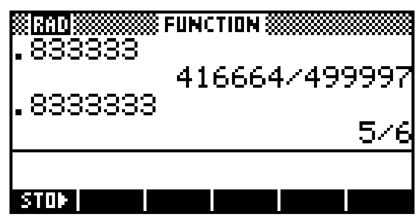
You may have noticed that all the results so far have been improper fractions. For example the first calculation shown right gives the answer as 22/15 rather than $1\frac{7}{15}$. The fraction setting of **Mixed Fraction** is essentially the same but answers are given as mixed fractions instead of improper fractions, as shown.



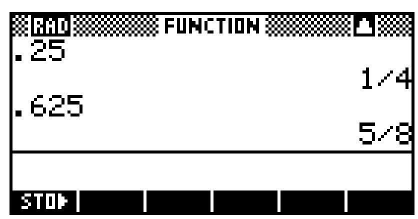
If you want to use the **Fraction** setting to convert decimals to fractions, here are some tips…

- if converting a recurring decimal to a fraction, then make sure you include at least one more digit in the decimal than the setting of **Fraction** in **MODES**. As you can see right, failing to include enough decimal places does not produce the desired result.



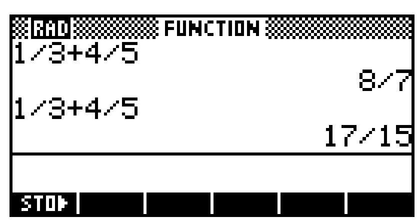


- if you are converting an exact decimal to a fraction, then set a **Fraction n** value of at least one more than the number of decimal places in the value entered. Both examples in the third screen shot to the right were done at **Fraction 6**.



Not understanding the significance of the setting of **Fraction** can produce some unfortunate effects. For example, at Fraction **2**, the value of **123.456** becomes **123**, with the 0.456 dropped entirely.

An example of this is shown right. If you use a setting of only **Fraction 2** to perform the calculation shown, you will find to your amazement that **1/3 + 4/5 = 8/7** , whereas using **Fraction 6** gives the correct answer.

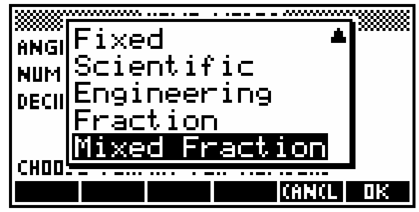


The reason for this 'error' is that the **1/3** and **4/5** were converted to decimals and added to give **1.133333**…. This was converted back to a fraction using **Fraction 2** to give **8/7** (**1.1428..**).

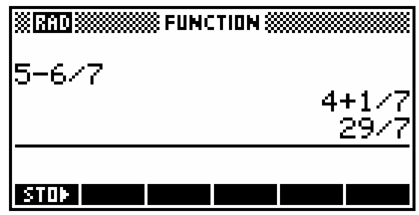
This may seem odd but it does match sufficiently closely in **Fraction 2** to be accepted.

Generally it is not a good idea to go below the default setting of **Fraction 4**. In fact, a **Fraction 6** setting tends to be more reliable.

A new feature of the hp 39gs and hp 40gs is the setting of **Mixed Fraction** in the **MODES** view.



The results of this new setting can be seen in the image to the right. Using the setting of **Mixed Fraction** the result is **4+1/7** ($4\frac{1}{7}$) whereas the answer of **29/7** is obtained using the old **Fraction** setting.



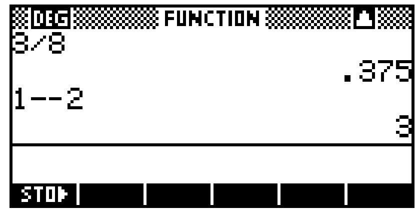
Calculator Tip

If you scroll back through the History and re-use a result such as the **4+1/7** shown above then don't forget to put brackets around it to ensure that no 'order of operations' errors occur.

The **HOME** *History*

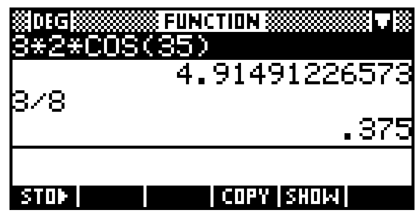
The **HOME** page maintains a record of all your calculations called the History. You can re-use any of the calculations or their results in subsequent calculations.

Try this for yourself now. Type in at least four calculations of any kind, pressing the **ENTER** key after each one to tell the calculator to perform the calculation.

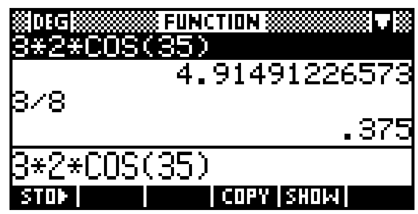


You will now be looking at a screen similar to the one on the right (except probably with different calculations).

If you now press the up arrow key, a highlighted bar moves up the screen. When you reach the top of the screen the previous calculations will scroll into view. Pressing **SHIFT** up-arrow takes the highlight to the top in one movement.



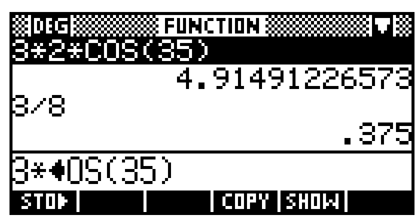
You may have noticed that as soon as the highlight appeared so did two labels at the bottom of the screen. If you now press the screen key under **COPY** you will find that the highlighted calculation will be copied on the edit line. This is shown in the screen shot on the right.



At this point you can use the left and right arrows and the **DEL** key to edit the calculation by removing some of the characters and/or adding to it.

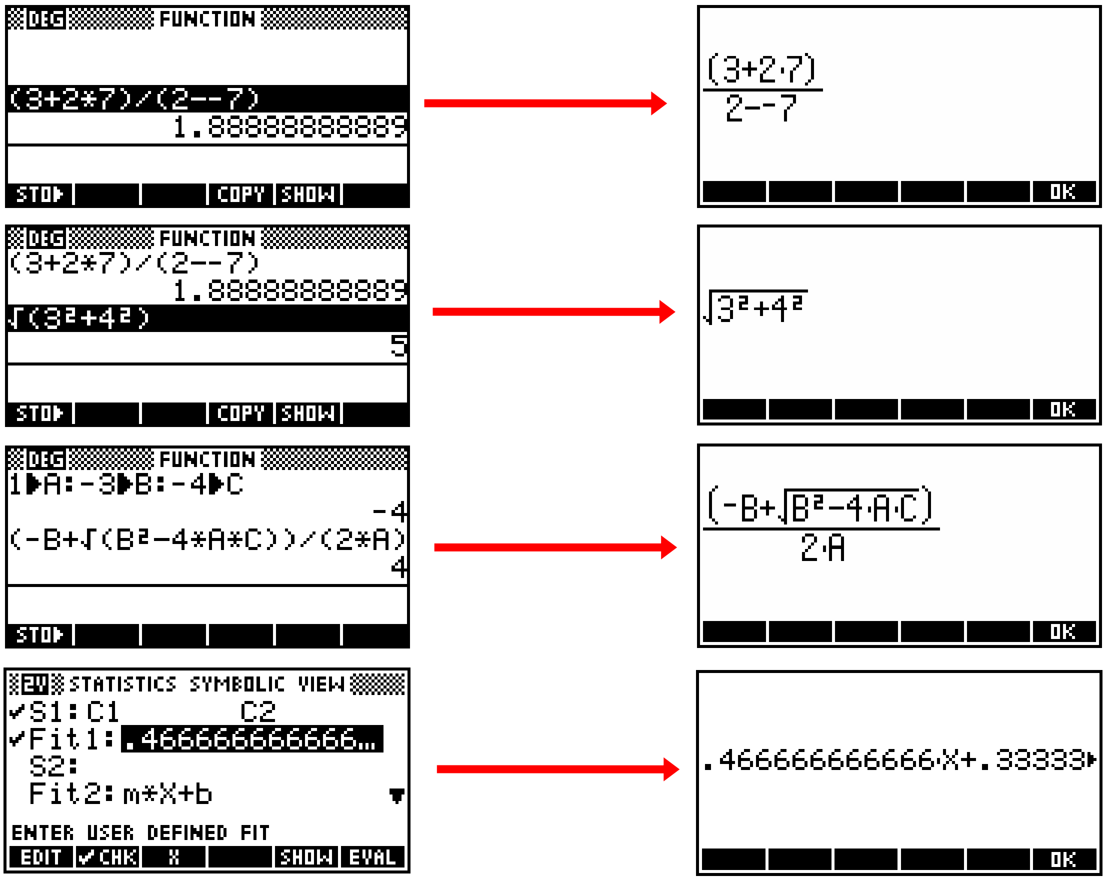
For example, in the screenshot right, the calculation of **3*2*COS(35)** has been edited to **3*COS(35)**.

Pressing enter will now cause this new calculation to be performed.



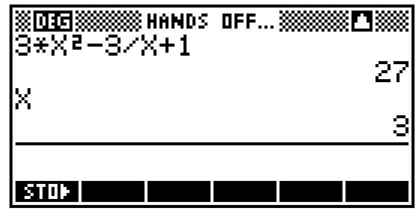
Calculator Tip
Pressing **ON** during editing will erase the whole line.
Pressing **SHIFT CLEAR** erases the whole history. This is worth doing regularly, since the history uses memory that may be needed for other things, even with the immense amount of user memory the hp 39gs & hp 40gs have. You can **COPY** calculations and results from any number of different lines in building your new expression.

Next to the COPY key you will see another screen key labeled SHOW. This key will display an expression the way you would write it on the page rather than in the somewhat difficult to read style that is forced on the calculator when it must show the whole expression on one line. This works anywhere the SHOW label appears, not just in **HOME**. Some examples…



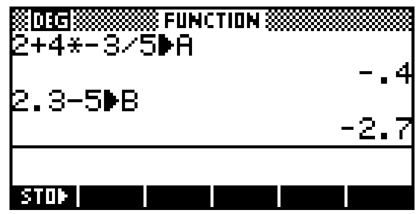
Storing and Retrieving Memories

Each of the alphabetic characters shown in orange below the keys can function as a memory. Some examples of this are shown in the third of the four examples above where the values of **1**, **-3** and **-4** are stored into **A**, **B** and **C** prior to the use of the quadratic formula. All of this storing of values is done with the **STO▶** key, which is one of the screen keys listed at the bottom of the **HOME** view. There are ways of obtaining even more memories than these 26 alphabetic ones, such as storing values into a list (see page 215), but 26 is enough for most purposes.



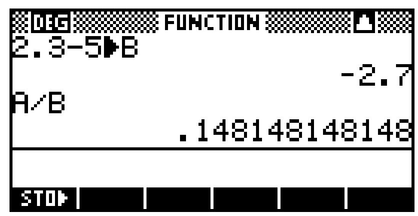
Once stored into memory, values can be used in calculations by typing the letter in place of the value. Typing a letter and pressing **ENTER** will display memory's contents. See right.

There is an advantage to storing results in memories, particularly if they are long decimals, or if you're going to be re-using the result a number of times.

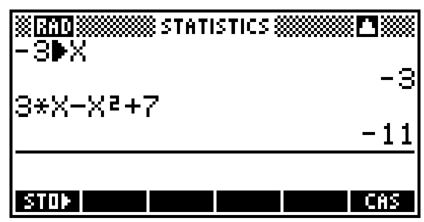


As an example, we will perform the calculation of $\dfrac{2+\dfrac{4\times(-3)}{5}}{2\cdot 3-5}$.

We will do this in two stages, calculating the top and bottom of the fraction and storing the results in memories.



Firstly the top of the fraction, storing the result in memory **A**, then the bottom, storing in **B**, and then finally the result…



As another example, suppose we were evaluating $3x-x^2+7$ for the value $x=-3$. We can store -3 into the memory X and then use that symbol in the calculation as shown right. The advantage of this is that if we now wanted to evaluate that expression for other values then we need only store a different value into X and then **COPY** the expression for re-evaluation. Clearly, if the expression is complex, this can be very helpful.

Calculator Tip

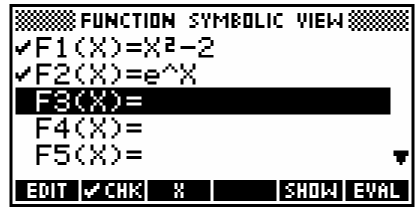
1. The memory **X** is the exception to the above information. It is used to store the current cursor position in the **PLOT** view so that you can access it in other places. Values stored in **X** will be overwritten the next time you use the **PLOT** view.

2. A common error is to forget to put brackets around negatives when evaluating expressions with powers. For example, evaluating x^2-1 as $-3^2 - 1$ rather than as $(-3)^2 - 1$. This can be avoided by storing the value into memory X and evaluating X^2-1. This is particularly useful in very complex expressions. Any memory can be used in this way, not just **X**.

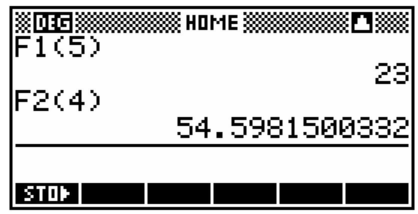
Referring to other aplets from the **HOME** *view.*

Once functions or sequences have been defined in other aplets, they can be referenced in the **HOME** view.

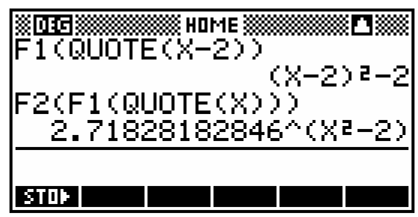
e.g. 1 Suppose we use the Function aplet to define **F1(X)=X²-2** and **F2(X)=eˣ** as shown right.



These functions now become accessible not only from within the **HOME** view but also within any other aplet also. This is shown by the screen shots below.

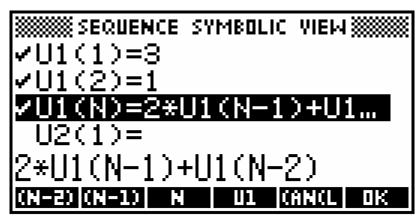


The results shown will, of course, depend on your settings in the **MODES** view.



The reason for the **QUOTE(X-2)** rather than just **X-2**, is that using **X-2** would tell the calculator to use the value currently stored in memory **X**, while **QUOTE(X-2)** tells it to use the symbol. The **QUOTE** function is available through the **MATH** menu under Symbolic (see page 181).

Note: This type of work is actually *far* more easily done in the Function aplet, where **QUOTE** is not needed and the **EVAL** key does a better job. See page 64.



e.g. 2 Suppose we use the Sequence aplet to define a sequence with $T_1 = 3, T_2 = 1$ and $T_n = 2T_{n-1} + T_{n-2}$.

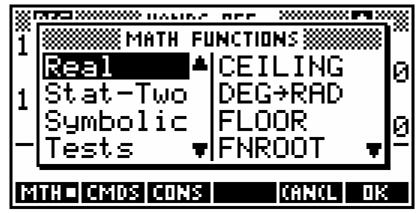
In the **HOME** view, the sequence values can be referred to as easily as the function values in the previous example.

You could also define a function **F1(X)** in the **Function** aplet and then refer to this function in the **Solve** aplet to find, for example, where **F1(X)=1000**.

A brief introduction to the MATH Menu

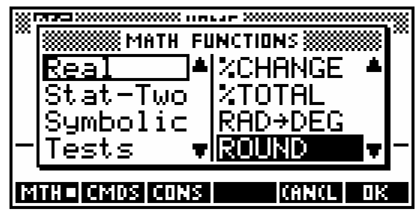
The **MATH** menu holds all the functions that are not used often enough to be worth a key of their own. There is a very large supply of functions available, many of them extremely powerful, listed in their own chapter beginning on page 165.

When you press the **MATH** key you will see the pop up screen shown right. The left hand menu is a list of topics. Scroll through the topics until you find the one you want, then use the right arrow key to move into the list of functions for that topic.



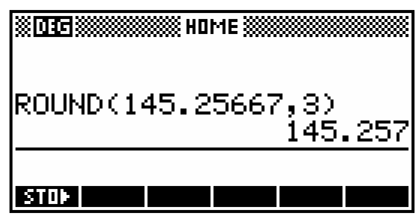
For example:

The function **ROUND** will round off to a specified number of decimal places.



E.g. round off 145.25667 to 3 decimal places.

Press the **MATH** key & right arrow (▶) into the **Real** group of functions.



Press the '**R**' key (the **9** key) to move to the first of the functions beginning with 'R', then move down one more function to **ROUND**. Press **ENTER**.

Now finish off the command and press **ENTER** again to see the result.

For more information on the Math Menu, see page 165.

Resetting the calculator

It is probably inevitable as the line between calculators and computers becomes blurred that calculators will inherit one of the more frustrating characteristics of computers: they crash! The hp 39gs and hp 40gs can sometimes do this due to the very complex and flexible operating system they use. If you find that the calculator is beginning to behave strangely, or is locking up then there are a number of ways to deal with this.

> **Calculator Tip**
> If you are a user of external aplets then you may occasionally find that an aplet that has been working perfectly will unexpectedly stop working with the message "Invalid syntax. Edit program?". There is almost certainly nothing wrong. Press **NO**, try a soft reboot as below, then run the aplet again.

Soft reboot (Keyboard)

Pressing **ON+SK3** will perform a soft reboot. It is perfectly safe and will not cause any memory loss except that your **HOME** History will be cleared. Hold down the **ON** key and, while still holding it down, press the third screen key from the left. The calculator will very briefly display a boot screen and then redisplay the **HOME** view, with the Function aplet as the active one.

If you find that the calculator locks up so completely that the keyboard will not respond then a method of reset is provided below which is independent of the keyboard. This shouldn't happen but it is important to know how to deal with it in case it happened during a test or an exam.

Soft reboot (Hardware)

Thi s method is provided in case the calculator is locked up to the point that the keyboard no longer responds.

On the back of the calculator is a small hole. Poke a paper clip or a pin into this hole and press gently on the switch inside. This briefly interrupts the power supply and, to the calculator, is exactly equivalent to a soft reboot as outlined in the previous section. Normally it should not only unlock the calculator but retain your data intact. However, it must be pointed out that a problem so severe that it has locked up the keyboard may be so bad that it has corrupted the memory anyway. If so then you may find that you see the "Memory Clear" message when it reboots. If you find that it keeps locking up repeatedly then you should perform a hard reboot as described in the next section. If you have important data on the calculator then you might try to save it to a PC before doing this since a hard reboot will wipe all user memory.

Hard reboot (with loss of memory)

To completely reset the calculator's memory back to factory settings press **ON+SK1+SK6**. (**SK1**="screen key 1") When doing this, don't press them all at once; hold down the **ON** key and, while still holding it down, press the first and then the last screen keys. Release them in the opposite order. Don't release **SK1** and **SK6** together - release **SK6**, then **SK1**, then **ON**. This is deliberately made complex so that it won't happen by accident!

This type of reset will always cause complete loss of data. If you find that the screen fills with garbage, or if the calculator's in-built diagnostic routine starts to run, then it is just that you have not released them in the right order. Simply try again.

Calculator Tip

1. If your calculator is really thoroughly locked up or won't turn on then try the procedures listed below this tip.
2. My past experience with these calculators is that they do tend to lock up occasionally, particularly if you load aplets and programs from the web. I usually suggest to my students that they regularly save their work to a PC and perform a hard reboot about once a month, reloading the saved work onto the calculator afterwards.
3. Pressing **ON+SK4** by mistake will result in the screen shown right. This is a special diagnostic mode that can be used by engineers to test the calculator's components. There is nothing wrong with experimenting with it but it is of no use to the average user. To exit, simply reboot normally by pressing **ON+SK3**.

```
1.LCD        2.KEY
3.FLASH      4.SRAM
5.USB        6.RS232
7.BUZZER     A.AUTOTEST
```

Locked Calculator? Won't turn on? Turns on briefly and then immediately off?

Try the following:

- Try a soft reboot using **ON**+**SK3**.
- Try a paperclip in the hole in the back (gently).
- Is it possible that the screen is blank because the contrast is turned to extremely light?
  Try **ON** and [+] to make the screen darker. (Hold down **ON** and press [+] repeatedly.)
  Use **ON** and [−] to lighten the screen.
- Change the batteries for new ones. Try this more than once because sometimes batteries are dead right off the shelf if they've been in the shop too long. Try replacing the backup battery too. Make SURE you've put the batteries in right way around. Putting them in reversed may seriously damage the calculator.

- Take the batteries out, including the round backup battery. Press and hold the **ON** button for 2 minutes to remove any possible remaining power from the internal capacitors. Leave the calculator overnight and re-insert fresh batteries.
- If this happened when running a game or aplet that you've downloaded from the internet then consider that this may be the source of the problem. Backup anything that you want to keep to the PC and do a full reboot to restore factory settings.
- Try a full reboot using **ON**+**SK1**+**SK6**.

Some less likely options:

- Did the paperclip feel funny when you inserted it? There should be a very subtle sensation of 'pressing a button' as the paperclip shorts out the batteries momentarily. Could you have accidentally bent the shorting contact last time you used a paperclip so that it is permanently shorting out?
- Check the battery compartment. Are any of the metal contacts broken or out of place? Are the batteries firmly in contact with all of them? Is anything out of place? Do any of the batteries or the metal contacts show corrosion? If so, clean them carefully, being careful not to get moisture inside the calculator.
- Have you recently dropped the calculator or spilled liquid on it? If so, this is not good. It is probably permanently dead. Did you need a very expensive paperweight?
- Check the USB and Serial ports at the top of the calculator. Do they have anything wedged in them courtesy of a young child perhaps?

Summary

- The up/down arrow key moves the history highlight through the record of previous calculations. When the highlight is visible, the **COPY** key can be used to retrieve any earlier results for editing using the left/right arrow keys and the **DEL** key.

- Care must be taken to ensure the your idea of order of operations agrees with the calculator's. For example, $(-5)^2$ must be entered as $(-5)^2$ rather than as -5^2, and $\sqrt{5+4}$ must be entered as √(5+4) rather than √5 + 4.

- The **ANS** key can be used to retrieve the results of the calculation immediately preceding the one you're working on. E.g. √**(5+Ans)**

- The **DEL** key can be used to erase single characters in the editing line or single lines in the **HOME** history. The **SHIFT CLEAR** key will delete the entire history. Regular clearing will ensure that memory is not gradually eroded.

- The **SHOW** key displays a calculation as you would see it written and is available in many views.

- The **MODES** view can be used to set the format in which numbers are displayed on the **HOME** page, and to choose the angle measure which is to be used.

- Make sure you understand **Fraction** mode before using it.

- Remember that the angle and numeric mode settings may change if you change aplets in the **APLET** view.

- Numbers are stored in memory using the key labeled **STO▸**. The stored values can then be used by simply putting the letter in the expression in place of the number.

- You can easily reboot the calculator if it locks up, generally without loss of memory. Make sure you know how to do this in case it happens during a test or an examination.

- Regularly saving your information to a PC will ensure that you don't lose anything important. Additionally, it is a good idea to completely reset the calculator occasionally so that any instability in the operating system has no chance to grow.

- Many extra functions are available via the **MATH** menu. For more information on the complete set of mathematical functions available in the **HOME** view (and anywhere else) see the chapter "The MATH menu" on page 165.

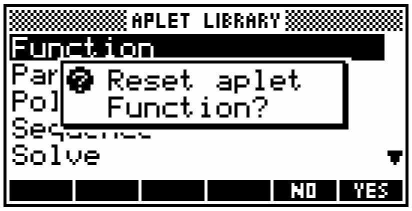
THE FUNCTION APLET

The Function aplet is probably the one that you will use most of all. It allows you to:

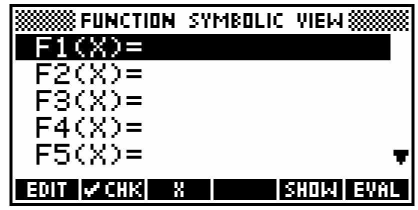
- graph equations
- find intercepts
- find turning points (maxima/minima)
- find areas under curves
- find areas between curves
- find gradients
- find derivatives algebraically
- find simple integrals algebraically
- evaluate functions at particular values
- graph and evaluate algebraically expressions such as f(g(x)) or f(x+2)

The first step for any aplet is to choose it in the **APLET LIBRARY**. Press the **APLET** key and you will see something similar to the screen on the right. Use the arrow keys to move the highlight up or down until the Function aplet is selected. Now, looking at the list of context sensitive functions at the bottom of the screen, you should see labels of START and RESET.

Press the key under RESET first. You'll see the message shown right - press the YES key. The reason for doing this is to clear out any functions that you may have put in there while playing around, and so to make sure that what you see will be the same as the screen snapshots.



Now press the **START** key. When you do, your screen should change so that it appears like the one on the right. This is the **SYMB** view. Notice that the screen title is supplied so that you will know where you are (if you didn't already).



Calculator Tip
Pressing **ENTER** rather than **START** would have had the same effect. Whenever there is an obvious choice pressing **ENTER** will usually produce the desired effect.

The XTθ button

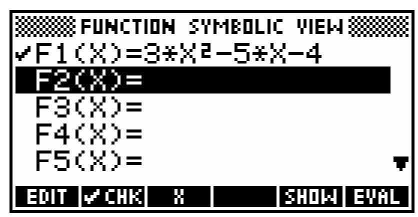
Whenever you enter an aplet, one of the keys which often changes its function is the key labeled **XTθ**. As its label suggests, it supplies an **X**, or a **T**, or a θ, depending on which aplet you are in. Let's use that key to produce a graph of the quadratic we dealt with in the earlier section on the **HOME** view.

Using the up/down arrows, move the cursor (if necessary) to the line labeled **F1(X) =** .

Type in:

[3] [X,T,θ] [x²] [–] [5] [X,T,θ] [–] [4] [ENTER]

This will produce the screen shown on the right. Notice the check/tick mark next to the **F1(X)** that appeared when you pressed **ENTER**. This signifies that this function is to be graphed, so that if you had five functions entered but only wanted 1 and 3 graphed, you could simply ensure that only **F1(X)** and **F3(X)** were checked.

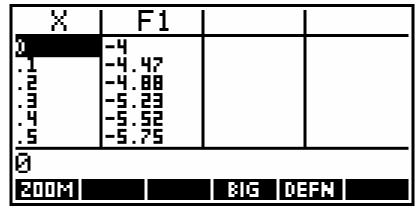


✔CHK*ing your function*

The key that turns the check on or off is a screen key - the one labeled with the ✔CHK. In computer terminology it is a 'toggle' *key* - when the check is off it turns it on and when on it turns it off. In some countries a check mark is called a 'tick' but the idea is the same.

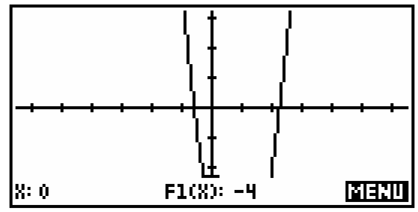
Try turning the check on and off for function **F1(X)**. Remember, the highlight has to be on the function before the check can be changed. Make sure it's checked when you have finished.

If you now press the **NUM** key, you will see the screen on the right. It shows the calculated function values for **F1(X)**, starting at zero and increasing in steps of **0.1**



Make sure the highlight is in the **X** column as shown on the previous page, and then press **4** and **ENTER**. You will find that the numbers will now start at four instead of zero. It is also possible to change the step size via the **ZOOM** key, which can be convenient at times, particularly for trig functions. The simplest way to set the start value and increment is in the **NUM SETUP** view. This is covered in detail on page 70.

Now press the **PLOT** key. The graph you'll see will not be a terribly useful one (see right) because the axes will not be set up correctly. We'll look next at how to do this.



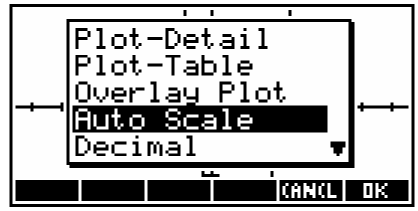
One of the easiest ways to set up the axes properly for a function whose shape is not known in advance is to let the calculator suggest a suitable scale using the *Auto Scale* option in the **VIEWS** menu covered on the next page. However, its suggested scales are not always very helpful. There are a number of ways of working out a 'nice' scale. See page 78 for a discussion of this. Another method is to use the **NUM** view to investigate the range of values that will be required for the function.

More complete information on the **VIEWS** menu can be found on page 85.

Auto Scale

Auto Scale attempts to fit the best possible vertical scale to the horizontal scale you have chosen. It is not always successful but will often give you a good starting point from which to refine the scale. In the example which follows it is assumed that you have plotted the graph as shown on the previous page.

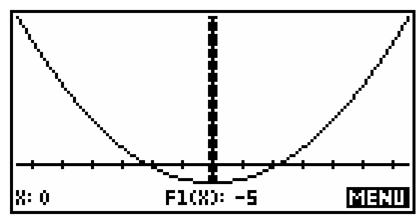
In the **PLOT** view, press the **VIEWS** key. Use the arrow keys to scroll down to *Auto Scale* and press **ENTER**. The calculator will adjust the y axis in an attempt to fit as much of the graph on to the screen as possible.



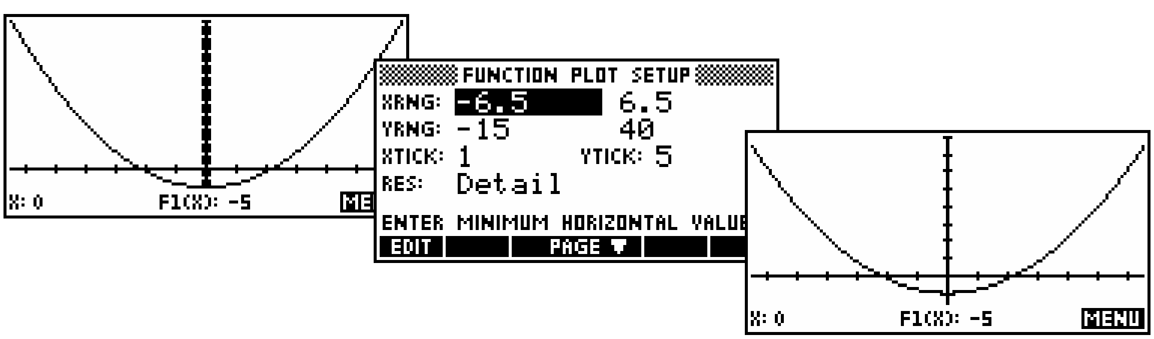
Some points to bear in mind;

- the y axis is scaled <u>only</u> on the first function which has a ✓CHK.
- the y axis is scaled for the current x axis (domain) you have chosen in **PLOT SETUP**. If you've not chosen the domain wisely then your result may not be good.
- it doesn't choose 'nice' scales for the y axis such as we would choose (going up in 0.2's or 5's or 10's etc.) so you will generally need to tidy up its choice a little.

If you look at the y axis of the graph you've just produced, you'll see that the axis tick marks are so close together that it looks like a solid line. To tidy this up you must change to the **PLOT SETUP** view. If you look above the **PLOT, SYMB** and **NUM** keys you will see the word '**SETUP**'. The **SETUP** view for each of these keys is obtained via the **SHIFT** key.

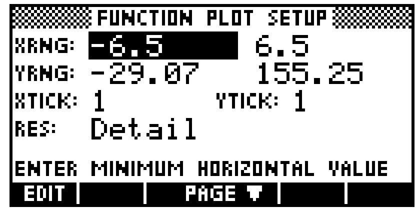


The values produced by the Auto Scale are shown below, along with a better set that produces a clearer **PLOT** view. Note particularly the **YTick** setting that removes the 'thick' effect on the y axis.



See "The Expert" chapter beginning on page 62 for more information on how to find good choices for axes. The Auto Scale function is also covered on page 89.

The PLOT SETUP view

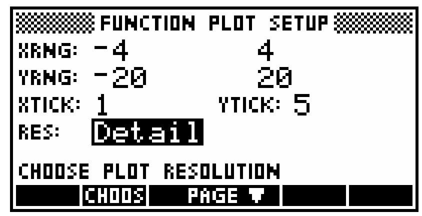


In the information that follows it will be assumed that you have performed the tasks on the previous page.

If you press **SHIFT** then **PLOT** you will see something like the view on the right.

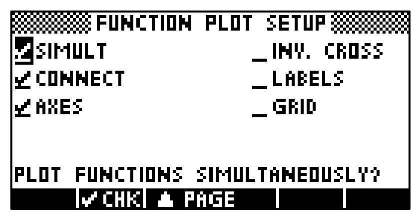
The highlight should be on the first value of *XRng*. Enter the value **-4**.

Calculator Tip
Don't use the subtract key to enter a negative. You MUST use the negative key labeled **(-)**. Similarly, don't use a **(-)** when you mean a subtract. For example, **2 (-) X** will produce **-2*X** not **2 – X**.

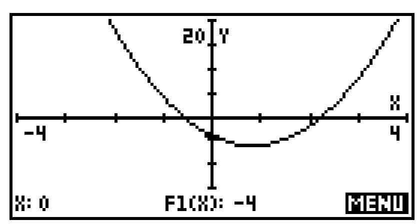


Type in **4** for the other **XRng** value, then **-20** and **20** for the **YRng** values. When you've done this use the arrow keys move to **Ytick** and change it to **5**.

At the bottom of the screen you will see **Res** (short for 'Resolution'). If you highlight it and press the **CHOOS** key you will see that you have a choice of *'Faster'* or *'More Detail'*. *'More Detail'* should be selected. If you choose *Faster* then every second dot is plotted instead of every dot. This is quicker but may make some graphs appear less smooth, particularly graphs with steep gradients.



There are two pages to this view (see the **PAGE ▼** key at the bottom of the screen). The first page is used to set axes, the second to control certain features of them.

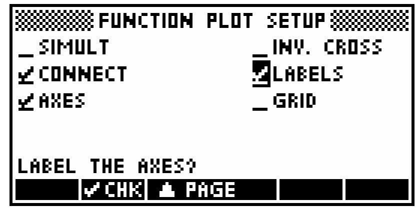


If you press the **PAGE ▼** key you will now be looking at the screen shown above right. Using the arrow keys to move the highlight, make sure that your checks/ticks match the ones in my snapshot. Now press **PLOT** again.

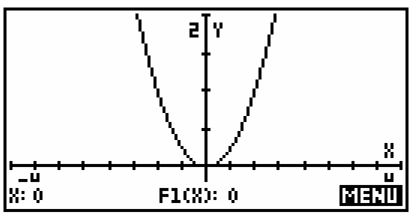
Let's have a look at the meaning of the CHKs (check marks or ticks) on the second page of **PLOT SETUP** above right. Although they are not used often they can be quite useful and I recommend highly that you at least consider using the *Simult* setting.

The first option **Simult** controls whether each graph is drawn separately (one after the other) or whether they are all drawn at the same time, sweeping from left to right on the screen.

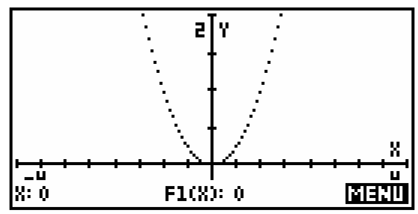
My preference is to turn this off. I find that if there are more than two functions defined then drawing them all at the same time can be confusing. Turning off **Simult** means that they are drawn one after the other, in the order that they are defined. This is obviously a bit slower but makes it easier to understand.



The second option **Connect** controls whether the separate dots that make up a graph are connected with lines or left as dots. This is very seldom of use.

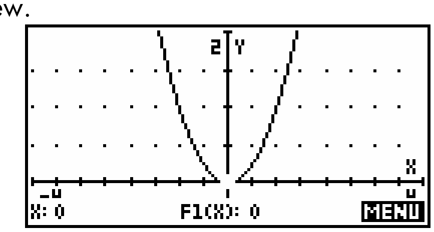


vs…



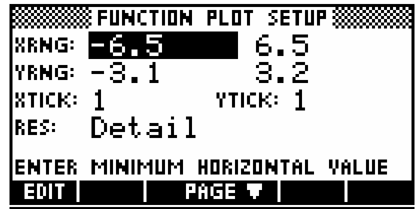
The third option **Axes** controls whether axes are drawn. The fourth **Inv. Cross** controls the appearance of the cursor that is moved by the arrow keys. It is best if you try this one yourself to see the effect.

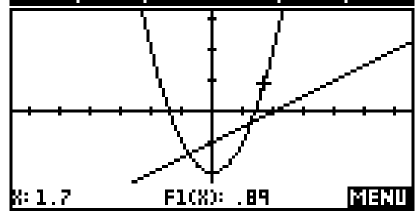
The fifth option **Labels** controls whether labels ('X', 'Y') and a scale are put on the axes. The only time this causes problems is if the scale is an odd one, causing the labels to have too many decimal places. If desired this effect can be offset by setting fixed decimal places in the **MODES** view.



The sixth and last option **Grid** causes a grid of dots to be drawn on the screen (see right). The density of the grid is controlled by the values of **Xtick** and **Ytick**. This can be quite useful.

The default axis settings

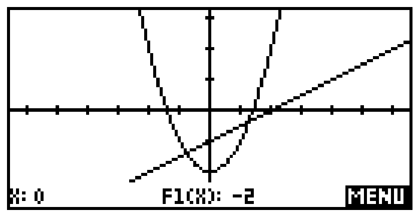




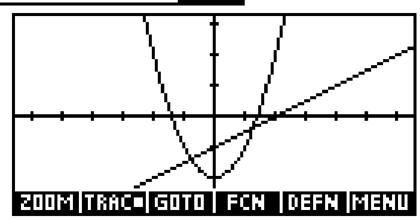
The default scale is displayed in the **PLOT SETUP** view shown right. It may seem a strange choice for axes but it reflects the physical properties of the LCD screen, which is 131 pixels wide by 63 pixels tall. A 'pixel' is a 'picture element' and means a dot on the screen.

The default scale means that each dot represents a 'jump' in the scale of 0.1 when tracing graphs. The y value is determined by the graph, of course, and has nothing to do with your choice of scale. Once the scale changes, the size of the jumps from dot to dot are often not a useful size. See page 62 for information on choosing 'nice' scales.

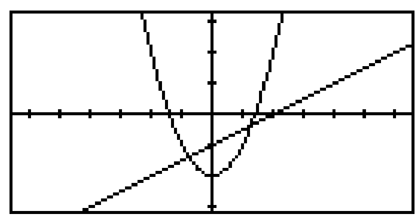
The **MENU** *Bar*



If you look at the screen key list at the bottom of the screen you will see only a single entry, labeled **MENU**.



Press the key under it and your screen will change to look like the middle one of the three right.

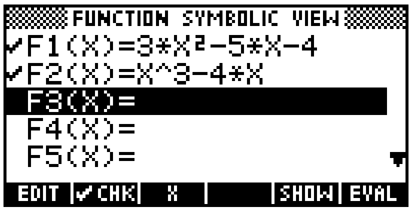


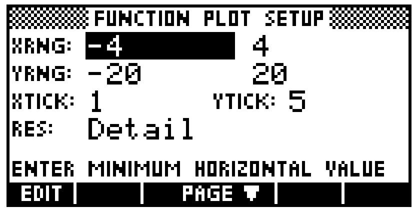
Press it again and the screen will clear completely. Once more and you are back to the original appearance. Try pressing it a few times to get the feel for its behavior. This is what is known as a 'toggle' switch. The **MENU** key is a triple toggle, cycling through each of the display modes shown above and right. The first (default) mode is (X,Y) mode, where the coordinates of the current cursor position are displayed at the bottom of the screen. In any of these modes the up/down arrows move the cursor from function to function, while the left/right arrows move along the currently selected function.

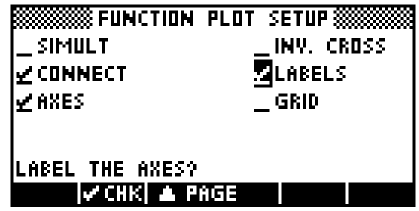
> **Calculator Tip**
> Pressing **SHIFT** right arrow or **SHIFT** left arrow will jump the cursor directly to the right or left side of the screen.

The Menu Bar functions

In the examples and explanations which follow, the functions and settings used are:

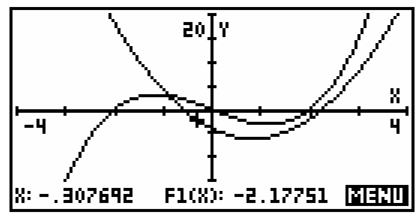






Trace

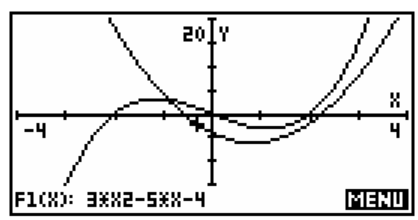
TRAC■ is quite a useful tool. The dot next to the word means that it is currently switched on. If yours shows **TRACE** instead then press the key underneath to turn it on. Leave it on for now.



Press the left arrow 5 or 6 times to see a similar display to that shown right. Pressing up or down arrow moves from function to function.

The order used when moving from graph to graph is not related to the physical location of the graphs on the screen but rather to the order that they are defined in the **SYMB** view. If **TRAC■** is turned off then the cursor is free to move anywhere on the screen.

Defn

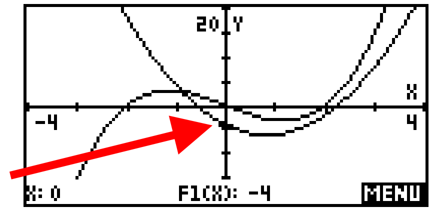


Press the key labeled **DEFN** (short for *Definition*). You will find that the equation is now listed at the bottom of the screen.

The up/down arrows will move the cursor from F1(X) to F2(X), with the definition changing as it does so. If **TRACE** is switched off then **DEFN** will not work correctly, nor will various other useful tools.

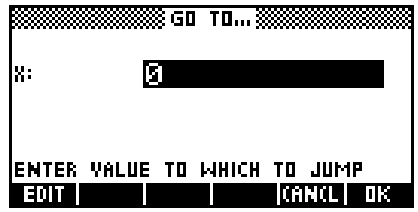
However, it does have the advantage that the cursor can be freely moved around the screen with the current coordinates displayed at the bottom of the screen. For this aspect to work properly you really need to choose a scale where the pixels are on 'nice' numbers. Multiples or fractions of the default scale are best for this.

Goto

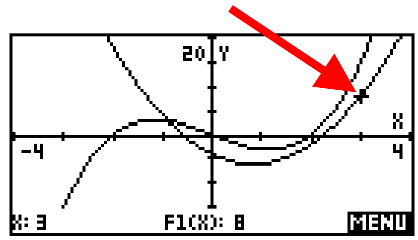


This function allows you to move directly to a point on the graph without having to trace along the graph. It is very powerful and useful.

Suppose we begin with the cursor at x = 0 on **F1(X)** as shown right.

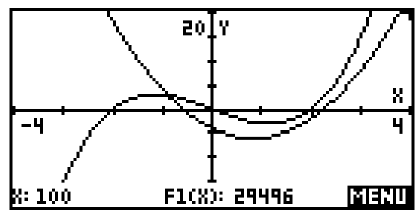


Press **MENU** and then **GOTO** to see the input form shown right.



Type the value 3 and press **ENTER**. The cursor will jump straight to the value x = 3, displaying the (X,Y) coordinates at the bottom of the screen.

A very nice feature of the **GOTO** key is that it will jump to values which are not on the current screen, or which would be inaccessible for the current scale.

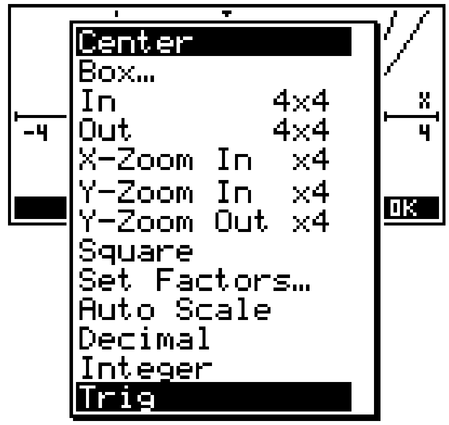


For example, we can jump to the value x = 100 and see the (X,Y) coordinate displayed, with the cursor positioned at the far right side of the screen. Similarly, you could jump to the value x = √2 despite this value being inaccessible for the scale chosen, since the cursor will normally only move to the values defined by the dots on the screen.

Calculator Tip

The **GOTO** key will also accept calculated values. You could, for example, jump to a value such as e^2+2. If you had recently found an intersection, then jumping to a value of **Isect** would return the cursor to that point. This is useful when finding areas between functions. See page 58 for information on **Isect**. See page 70 for an example of finding areas between curves using **GOTO**.

The Zoom Sub-menu



The next menu key we'll examine is **ZOOM**. Pressing the key under **ZOOM** pops up a new menu, shown right.

The list which follows covers the purpose of the first nine options shown right, down to "Set Factors".

The four final options which follow these are also on the **VIEWS** menu and are covered on page 85 as part of the detailed examination of the **VIEWS** menu.

Center

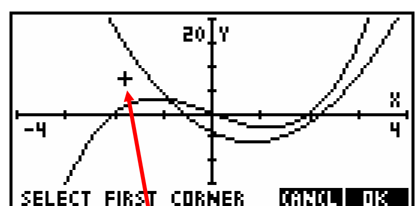
This redraws the graph with proportionally the same scale for each axis but re-centered around the current position of the cursor. If you already have a 'nice' scale, this will preserve it, while perhaps showing a more interesting section of graph.

In/Out

These two options zoom in or out by adjusting the scales by the factor shown. The default factor is 4 for both axes but this can be adjusted through the *Set Zoom Factor* option later in the menu. The most useful settings are either 5x5 or 2x2 as these are more likely to preserve nice scales.

Box…

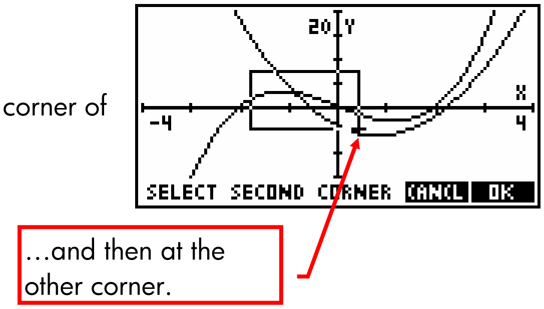
This is the most useful of the **ZOOM** commands. When you choose this option a message will appear at the bottom of the screen asking you to *Select first corner*.



The cursor is positioned at one corner of a box.

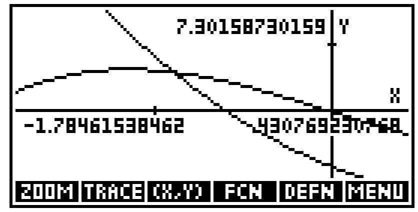
If you use the arrow keys to move the cursor to one corner of a rectangle containing the part of the graph you want to zoom into and then press **ENTER**, the message will change to *Select second corner*.

As you move the cursor to a position at the diagonally opposite corner of a rectangle, the selection box will appear on the screen.

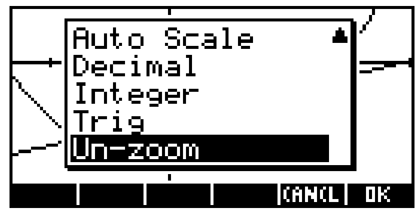


Pressing OK expands the box to fill the screen.

You'll notice that the scale has been disrupted so that the labels are no longer very helpful. **PLOT SETUP** would give better end points for the axes or let you switch off the labels option. Alternatively you could use the **MODES** view to set two decimal places.

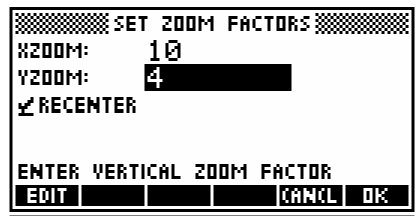


Rather than doing any of these however, scroll down the ZOOM menu to a new option of *UnZoom*. This option puts the screen back to the way it was before you did the ZOOM.

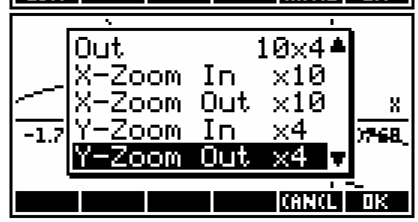


X-Zoom In/Out x4 and Y-Zoom In/Out x4

These two options allow you to zoom in (or out) by a factor of 4 on either axis.



The factors can be set using the *Set Factors…* option, which gives you access to the view shown above right. You will also see a check mark next to an option called **Recenter**. If this is ✓CHKed then the graph will be redrawn after zooming in or out with the current position of the cursor as its center.



Changing the x factor is reflected in the ZOOM menu as you can see in the second screen snapshot.

Square

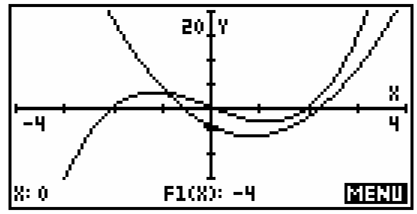
This option changes the vertical scale to proportionally match the horizontal scale. This will ensure that circles will appear properly circular rather than falsely elliptical.

Auto Scale, Decimal, Integer and Trig

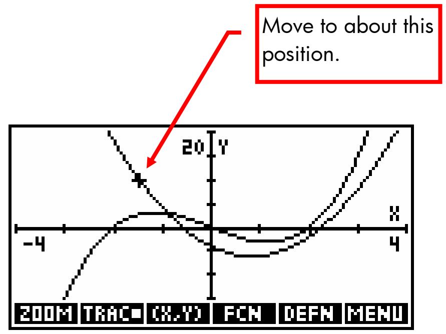
These four options are duplicates of those found on the **VIEWS** menu, and are provided in both places simply for convenience. For information on them, see the section on the **VIEWS** menu on pages 85.

The FCN menu

Note: *Before continuing, set the axes back to the way we set them at the top of page 53.*

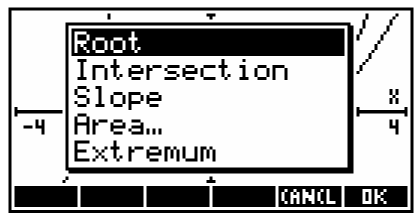


Looking at the menu functions again, you will see that the only one we have not yet examined is the one labeled **FCN**. This key pops up the Function Tools **FCN** menu and is the most useful of them all.

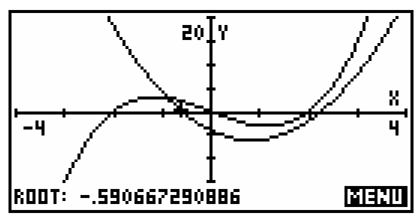


Before you use this key, make sure **TRACE** is switched on (ie showing **TRAC■**) and then move the cursor so that it is in roughly the position shown right.

Root



Press the **FCN** key. As you can see on the right, this key gives you access to a number of useful tools. If you leave the highlight on *Root* (as shown) and press **ENTER** (or **OK**) then the cursor will jump to the nearest root or x intercept for the function it is on, starting its search at the current position of the cursor. Try it now.

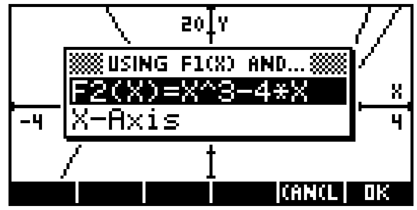


Notice the message at the bottom of the screen giving the value of the root that was found. To find the other root, you need to move the cursor so that it is closer to the other root than to the present one. In this case that means moving it past the turning point.

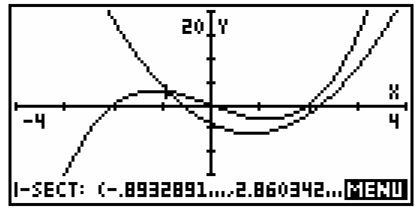
Calculator Tip
If you are working with a function which has asymptotes then make sure the cursor is positioned on the same side of the asymptote as the root. The internal algorithm used to find roots does not cope well with crossing asymptotes.

Intersection

The next function tool in the **FCN** menu is *Intersection*. If you choose this option, then you will be presented with a choice similar to the one in the screen shown right. The *Intersection* option only appears if there is more than one function in the **PLOT** view.



Exactly what is in the menu depends on how many functions you have showing. In the case shown here we only have two, so the choice is of finding the intersection of **F1(X)**, which is the one the cursor is on, with either the X axis, or the other function **F2(X)**. The results of choosing **F2(X)** are shown right.



Calculator Tip
When you find an intersection or a root the value of the x coordinate is stored in the memory **X**. If you immediately change to the **HOME** view and type **X** and hit **ENTER** then you can retrieve and use this value. See "The Expert" on page 75 for more detail and examples.

Slope

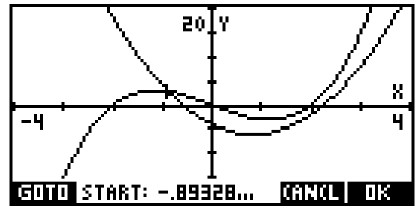
This gives the numerical value of the derivative at the point of the cursor for the current function. There are many other methods of doing this, some of which can be found on pages 66, 70 & 83.

Calculator Tip
If the value at which you wish to find the slope is not accessible for the current scale then use the **GOTO** key to jump to the desired point before choosing *Slope*. The value is remembered and used in the *Slope* calculation, despite perhaps not being 'on' the current scale. In the same way, choosing *Slope* immediately after finding a root or intersection will give the slope at the remembered value rather than the nearest dot on the scale.

Signed area…

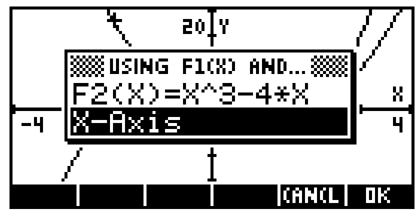
Another very useful tool provided in the **FCN** menu is the *Signed Area* tool. Before we begin to use it, make sure that **TRACE** is switched on, and that the cursor is on **F1(X)** - the quadratic. The *Signed Area* tool is similar to the Box Zoom in that it requires you to choose start and end points of the area to be calculated.

Suppose we want to find the definite integral: $\int_{-2}^{3} x^2 - 5x - 4\,dx$



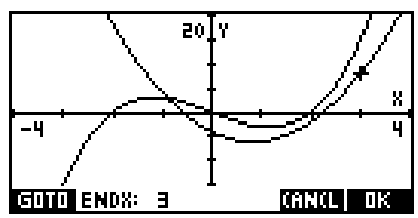
Choose **FCN** and then *Signed Area*. At the bottom of the screen you will see a prompt, as shown right, asking you to choose a starting point.

Press the **GOTO** key, enter the value **-2** and press **OK** or **ENTER**. The starting value will then be -2 so press **OK** again (or **ENTER**) to accept it.



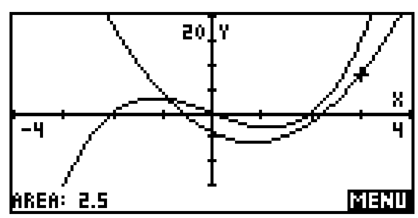
Another menu will now pop up, asking you to choose what area you wish to calculate. In this case there are only two choices: between **F1(X)** and the x axis, or between **F1(X)** and **F2(X)**.

If we had defined more functions in the **SYMB** view then this menu would be longer. In this case we want the area between **F1(X)** and the x axis, so position the highlight as shown and press **ENTER**.



The graphs will then reappear, with a message requesting that you choose an end point. In the screenshot shown right I have pressed **GOTO** and entered the value 3 to move to that point directly.

If you now press **ENTER** again to accept the end point, the hp 39gs or hp 40gs will calculate the signed area and display the result at the bottom of the screen.



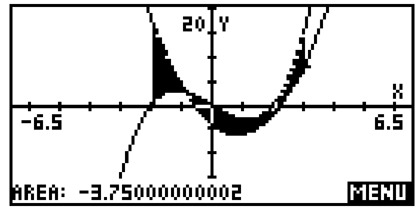
Calculator Tip

It should be clearly understand that although the label at the bottom of the screen is **Area** it is a little misleading.

What has actually been calculated is the definite integral (right), with 'areas' below the x axis included as negatives. This is why the label on the original menu reads *"Signed area"* instead of just *"Area"*.

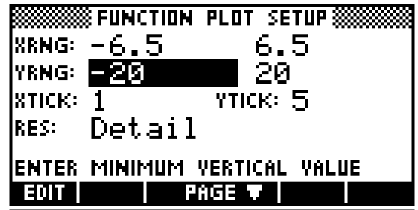
$$\int_{\text{1st point}}^{\text{2nd point}} F1(X)\,dx$$

Rather than using the GOTO key, an alternative method is to use the tracing facility. The advantage of this is that the 'area' is shown visually as you go by shading, as can be seen right.

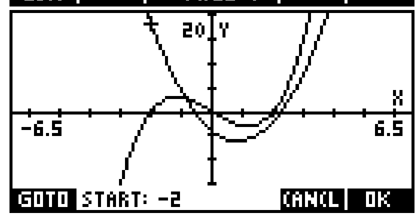


The disadvantage of this is that you can only trace to values which are permitted in the scale you are using. As soon as you use GOTO the shading stops. In this case, due to the scale we chose, if you try to trace to the values x = -3 or x = 2 you will find that they are not accessible. To illustrate the process we will need to change scale.

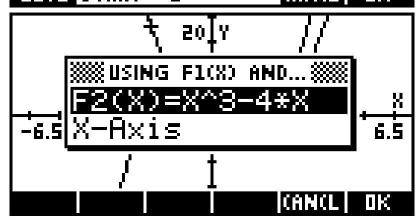
Change to the **PLOT SETUP** view and set the x axis (only) back to the default values **-6.5** to **6.5**, then **PLOT** again. This can be done by highlighting the **XRng** fields and pressing **DEL**.



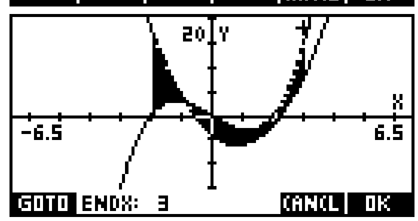
Press MENU and FCN again, choosing *Signed Area...* as before. Use the left/right arrow keys or the GOTO key to move the cursor to x = -2. Press **ENTER** to accept the starting point.

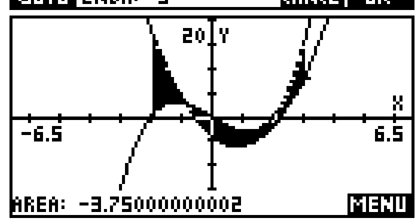


This time, choose the boundary as **F2(X)** instead of the x axis so that we will be finding the signed area between curves instead of the signed area under one. Again, note that the result will be a signed area (definite integral) not a true area. See page 75 for a simple method of finding true areas.



We now need to choose the end point. This time do it by tracing with right arrow to move the cursor. As you do the area will be shaded by the calculator. The current position is shown at the bottom of the screen. When you reach the end point you are looking for, press the OK key and the area will be calculated as before. This is shown right. To remove the shading, press **PLOT** again to force a re-drawing of the screen.





Calculator Tip

Common sense tells us that the answer in the example above is almost certainly -3.75 rather than -3.75000000002. The small error is simply due to accumulated rounding error in the internal methods used by the calculator. For example, an answer of 0.4999999999 should be read as 0.5 This is quite common and users should be aware of the need for common sense interpretation.

Areas between and under curves

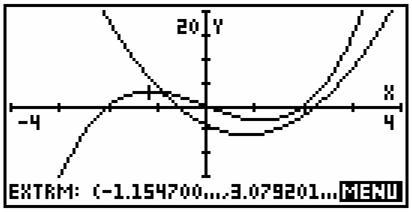
If we are wanting to find true areas rather than the 'signed areas' given by a simple definite integral then we must take into account any roots of the function.

The difficulty with this is that for most functions the roots and intersections are quite unlikely to be whole numbers and rounding them off will produce inaccuracies in the calculation.

There are ways to get around this and the process is shown in detail on page 75. Basically the trick is to find each root in turn and use the **HOME** view to store the values of the roots as **A**, **B**, **C**… These values can then be used in the calculation to retain full accuracy.

Extremum

The final item in the **FCN** menu is the *Extremum* tool. This is used to find relative maxima and minima for the graphs.



Ensure that **TRACE** is switched on and that the cursor is positioned on the cubic **F2(X)** in the vicinity of the left hand maximum (turning point) as shown right. Press **FCN** and choose *Extremum* from the menu. You should find that the cursor will jump to the position of the maximum.

Calculator Tip
If your graph has asymptotes then make sure that the cursor is positioned on the side of the asymptote containing the extremum before initiating the process. The internal algorithm used does not cope well with intervening asymptotes.

9

THE EXPERT: WORKING WITH FUNCTIONS EFFECTIVELY

Finding a suitable set of axes

This is probably the most frustrating aspect of graphical calculators for many users and there is unfortunately no simple answer. Part of the answer is to know your function – this is why we still expect you to learn mathematics instead of expecting the calculator to do it all! If you know, for example, that your function is hyperbolic then that immediately gives information about what to expect. If you don't have knowledge then here are a few tips:

- Try plotting the function on the default axes. You may find that enough of the function is showing to give you a rough idea of how to adjust them to display it better. Remember that **ZOOM** can work on either axis or on both. When in doubt, zoom out rather than in. See Tip #4 on the next page.

- The **NUM** view can be very helpful. Try changing to **NUM SETUP** and setting the value of **NumStep** to 5 or even 10. Now scroll through the **NUM** view and look at what is happening to the **F(X)** values. Look for two things.

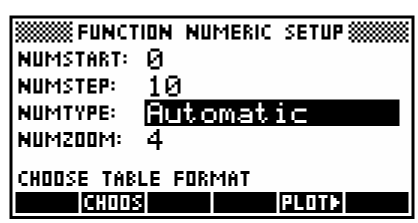
    - Firstly, where is the function most active? For what domain on the x axis is it changing rapidly both up and down? This is likely to be the domain you are most interested in.

    - Secondly, what is the range? What sort of values will you need to display on the y axis? Change to the **PLOT SETUP** view and set what you think may be appropriate axes. From those you can **PLOT** and then zoom in or out.

- If the graph is part of a test or an examination then the wording of the question will often give a clue as to what x axis domain you should work with. You can then use Auto Scale.

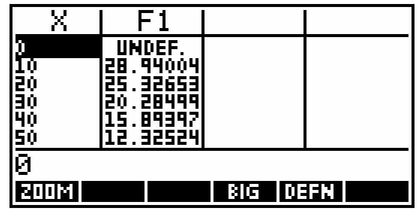
Auto Scale can be used to get a first approximation to a good set of axes. To do this you must choose your x axis domain first. Use your knowledge of what the function might look like, perhaps together with a quick scroll through the **NUM** view, to get an idea of what section of the x axis is important.

For example, suppose a company's profit is modeled by the equation $P(x) = \frac{-100}{x} + 50e^{-0.025x}$, where *x* is the number of items manufactured. Suppose that from the context of the question it is clear that we are mainly interested in the domain $0 \le x \le 100$.

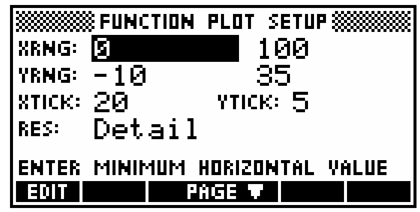


After entering the equation into the **SYMB** view, change into the **NUM SETUP** view and set the numeric scale to start at zero and increment in steps of 10 (or 5 if you want more details).

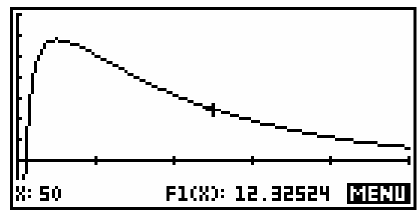
Change into the **NUM** view and scroll through the window from zero to 100. As you do so, take note of the values that the function takes. From the display it seems that the function peaks around y=30 and then declines steadily.



Change into the **PLOT SETUP** view and enter an x axis of 0 to 100 with a 'tick' value of 20, and a y axis of -10 to 35 with a tick value of 5.

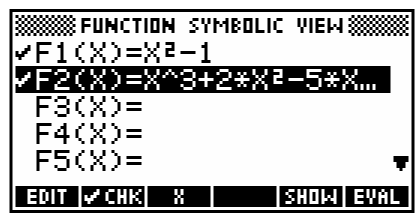


The result of this is a **PLOT** view as shown right. This would be ideal for answering questions on the domain stated.

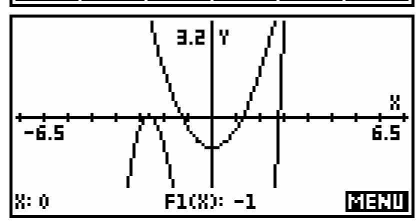


Another possible strategy for graphing which works quite well and, perhaps importantly, always gives 'nice' scales is to use **ZOOM**.

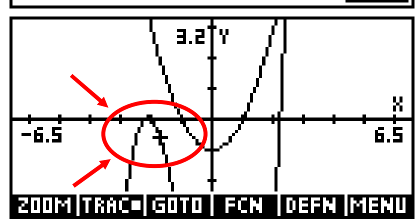
Enter your graphs into the **SYMB** view. Remember that *Auto Scale* only works on the first ticked graph.



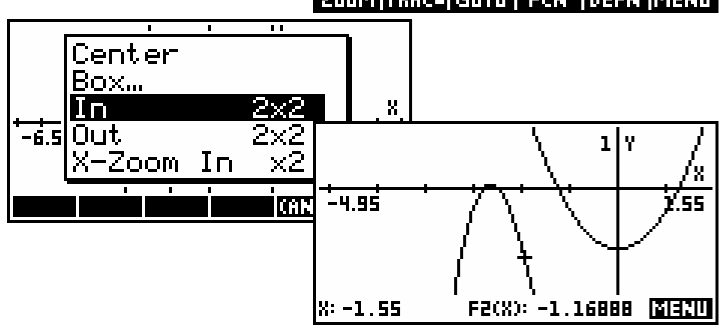
Press **VIEWS** and choose *Decimal*, or press **SHIFT CLEAR** in the **PLOT SETUP** view. This will give you the default axes, probably not showing the graph very well.



Place the cursor so that it is in the center of the area you are most interested in.



Use the **ZOOM** menu to adjust the view. You may choose first to change the zoom factors to something other than 4x4, and to ensure that *Recenter* is **✓CHK**ed. The **PLOT** view on the right is the result of setting 2x2 and re-center.



The advantage of doing it this way is that if you zoom in or out by a factor of 2 or 4 or 5, the cursor jumps will stay at (relatively) nice values allowing you to trace more easily. In this case, the cursor now moves in jumps of 0·05, which is ideal for most purposes. If you are not interested in tracing along the graph then this may not be important.

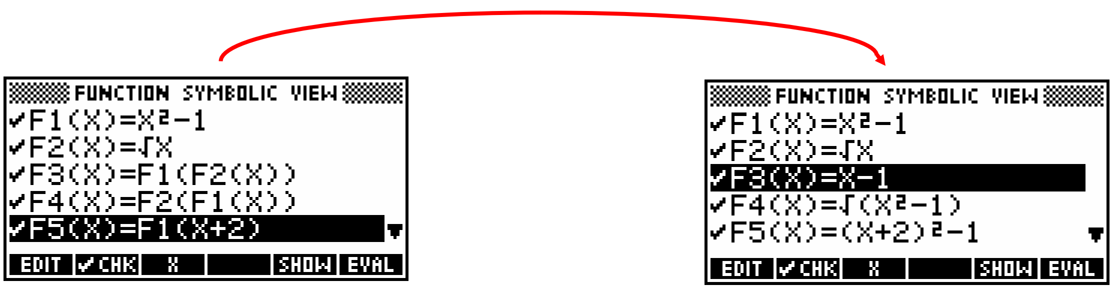
The disadvantage of this method is that you need to have at least some of the graph showing on the screen before you can zoom in or out to show more! Auto Scale can sometimes give you this first step.

Composite functions

The Function aplet is capable of dealing with composite functions such as $f\left(x+2\right)$ or $f\left(g\left(x\right)\right)$ in its **SYMB** view. The **EVAL** and **SHOW** keys are particularly helpful with this.

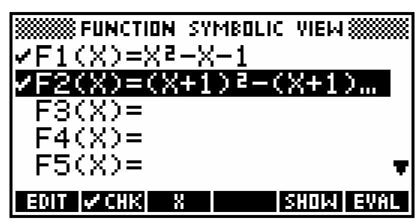
For example, if we define $F1(x)=x^2-1$ and $F2(x)=\sqrt{x}$, then we can use these in our defining of **F3(X)**, **F4(X)**. See the screen shot on the left below.

If the highlight is now positioned on each of these in turn, and the **EVAL** key pressed then the substitution is performed. The result is shown in the right hand snapshot.

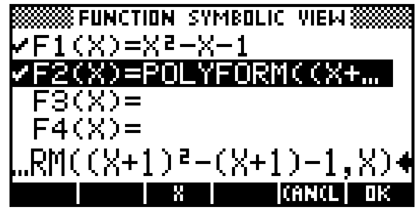


Notice that the calculator is smart enough to realize in **F3(X)** that $\left(\sqrt{x}\right)^2-1$ is the same as $x-1$, although not, unfortunately, smart enough to keep track of the implications for the domain, which are that **F3(X)** should be defined only for non-negative *x*.

There is a limit to this however. If you define $F1(x)=x^2-x-1$ and then $F2(x)=F1(x+1)$, then the **EVAL** routine will not simplify $\left(x+1\right)^2-\left(x+1\right)-1$ to x^2+x-1 .



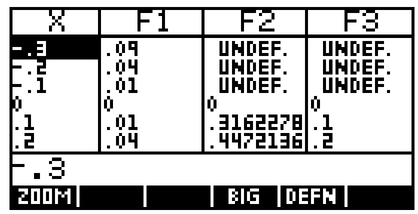
On the other hand there is a way to further simplify the expression. **EDIT** the result and enclose it with the **POLYFORM** function as shown right, adding a final '**,X**' as shown, then highlight it and press **EVAL**. The calculator will expand the brackets and gather terms.



Calculator Tip

These functions can all be graphed but the speed of graphing is slowed if you don't press **EVAL** first. This is because the *composite* function is internally re-evaluated for each point graphed. The hp 39gs and hp 40gs are fast enough that the result is still satisfactory but if you have an old 39g or 40g they are slowed to the point of being unusable.

Evaluating the function may also hide the domain and for this reason it is sometimes best to leave the evaluation undone.



For example, if $F1(X) = X^2$ and $F2(X) = \sqrt{X}$ then $F3(X) = F1(F2(X))$ will show the correct domain of $x \geq 0$ for both **F2(X)** and **F3(X)** in the **NUM** view. Pressing **EVAL** will destroy this.

Using functions in the HOME view

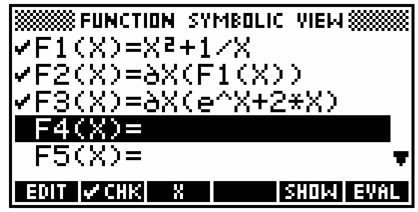
Once functions have been defined in the **SYMB** view of the Function aplet, they can be reused in the **HOME** view (and indeed in any other aplet!).

For example suppose you needed to find some exact points (x = 0, 1, 2 and 3) for a graph of $f(x) = \dfrac{2x}{(x-2)}$ when doing a hand sketch of it.

Type its definition into the Function aplet **SYMB** view, switch to the **HOME** view and then simply type **F1(0)** and press **ENTER**. The function will be evaluated for x = 0. Similar results can be obtained for **F1(1)**, **F1(2)** and **F1(3)**. Notice the error message for x = 2 caused by a divide by zero.

A simpler way to get function values is to changing into the **NUM** view and use its tabular listing.

Differentiating



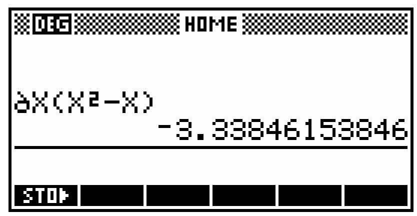
There are different approaches that can be taken to differentiating, most of which are best done in the **SYMB** view of the Function aplet.

The syntax of the differentiation function is:

$$\partial X(\,function)$$

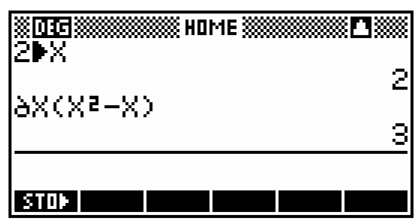
where *function* is defined in terms of X.

The function can be already defined in the **SYMB** view of the Function aplet as shown in functions **F1(X)** and **F2(X)** in the screen shot above. Alternatively it can be directly entered into the brackets as shown in function **F3(X)**.

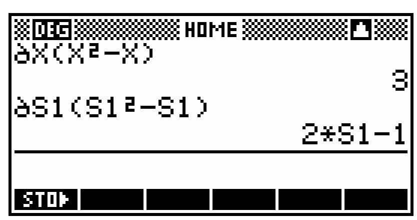


The ∂ symbol most easily obtained by pressing the key labeled **d/dx**. It can also be found in the **MATH** menu.

One point to remember is that if you use this function in the **HOME** view you may not receive the result you expect. If you try this yourself your result will probably not be the same as that shown right.

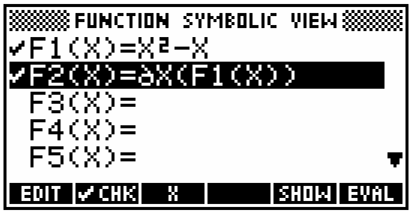


The reason for this is that the result you see is the derivative of $x^2 - x$ evaluated at whatever value of *x* happens to be currently in memory. This can be seen more clearly if we store a specific value into the memory **X** beforehand. In the example shown right, the answer of 3 is the value of the derivative $2x-1$ at the value of *x* = 2.

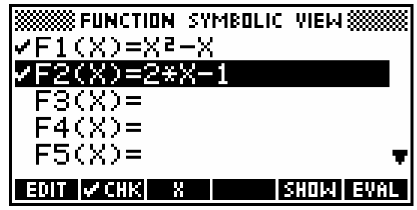


But what of algebraic differentiation? It is possible but not very convenient to do this in the **HOME** view using a "formal variable" of **S1**. The drawback of this is simply the awkwardness of having to work with **S1**'s rather than with **X**'s.

Algebraic differentiation is most easily handled in the **SYMB** view of Function. The best method is to define your function as **F1(X)** and its derivative as **F2(X)** (see below)…

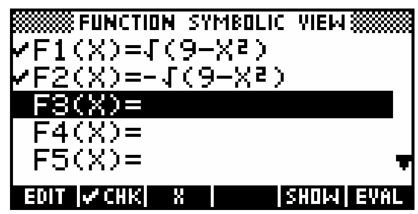


press EVAL



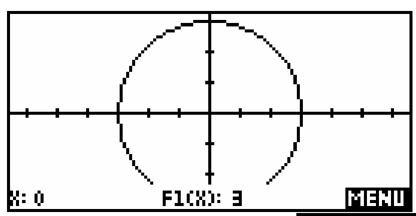
Calculator Tip
Doing your differentiation in the Function aplet is much easier and offers the additional advantage of being able to graph the two functions.

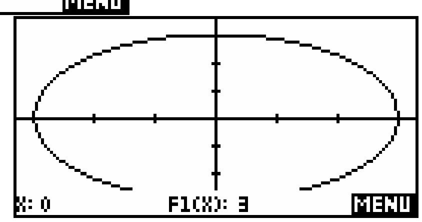
Circular functions

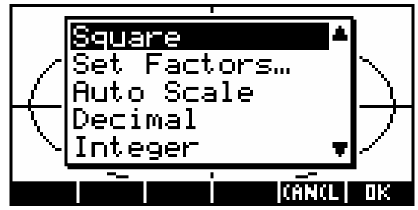


There are two issues that influence the graphing of circular functions, both related to the scale chosen.

The first one, illustrated on the right, causes circles to be ellipses if you don't choose scales for the x and y axes which are 'square' relative to each other. The two graphs above are both of the same function $x^2 + y^2 = 9$.

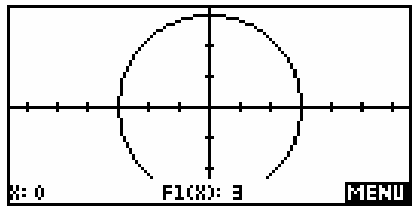




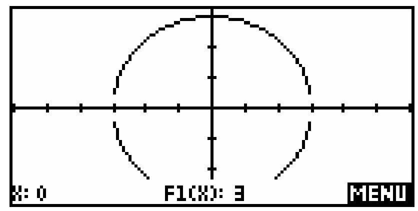


The simplest way to deal with this is to use scales which are multiples of the default scales. For example by using $-13 \le x \le 13$ and $-6.2 \le y \le 6.4$. These are a scale factor of 2 from the default axes of $-6.5 \le x \le 6.5$ and $-3.1 \le y \le 3.2$.

You can also use the *Square* option on the **ZOOM** menu. This adjusts the *y* axis so that it is 'square' relative to whatever *x* axis you have chosen.

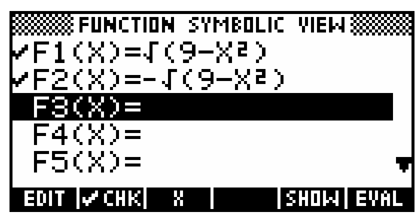


The second issue is caused by the domain of the circle being undefined for some values. The screen on your calculator is made up of small dots called pixels and is 131 pixels wide and 64 pixels high. As was mentioned earlier, this means that each pixel is 0.1 apart on the x and y axes.



This can affect your graphs and it becomes particularly obvious with circles because the graph does not exist for the part of the x axis outside the circle. The two screen shots right are an example of two images of the same graph $x^2 + y^2 = 9$ using two slightly different scales. You can see that the second example has missing pieces.

Let's look at the circle $x^2 + y^2 = 9$ as an example. This circle only exists from -3 to 3 on the x axis and is undefined outside this domain. In order to graph it you have to rearrange it into two equations of:



F1(X)= √(9-X²) for the top half
& **F2(X)= - √(9-X²)** for the bottom half.

If you enter these equations and then graph them with the default axes then you get a perfect circle. However, if you change the x axis to -6 to 6 rather that the default setting and then **PLOT** again you will find that part of the circle disappears. This is shown in the second snapshot above.

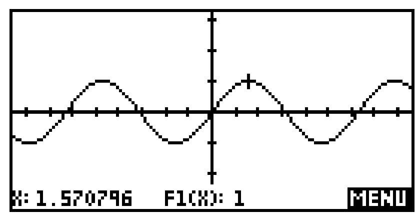
The reason for this is that when the calculator draws the graph it does so by 'joining the dots'. For the default scale of -6.5 to 6.5 this is not a problem since the edges of the two half circles at -3 and 3 fall on a pixel. This means that the last segment of the graph plotted extends is from 2.9 to 3 and the circle reaches right down to the x axis.

However, for the scale of -6 to 6 the pixels are no longer 'nice' values of 0.1. If you try to trace the circle you'll see that the pixels fall on 0, 0.0923077, 0.1846154..... In particular, near x=3 the pixel values are 2.953846 and 3.046154. This means that the calculator can't draw anything past 2.953846 because the next value doesn't exist, being outside the circle. This is what causes the gap in the circle. There's nothing to join to past that last point.

If this missing piece is a problem to you then the solution is to use scales which allow the end points of your circle to fall on a pixel point. If your circle has an integer radius then this is easily done by starting with the default axes and then **ZOOM** in or out to show the circle. This will tend to give 'nice' pixel points. If your circle is not centered on the origin then just check/tick the box in the *Set Zoom Factors* box to *Recenter*. That will allow you to turn off **TRAC■**, move the cursor closer to the point where you'd like the centre of screen to be and then **ZOOM**.

Trig functions

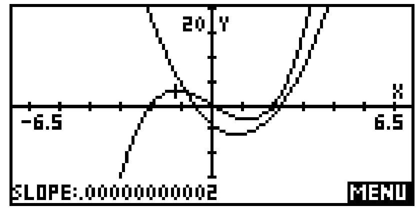
If you intend to **TRACE** the graph of a trig function and you are using radian measure then you should choose a scale which will cause pixel values (screen dots) to fall at convenient points. A good scale to use is provided on the **VIEWS** menu (see right).



This scale sets each dot to be $\pi/24$, ensuring that multiples and most factors of π will fall on dots and thus helping if you want to trace values on the graph. You can see in the example right that the value of $\pi/2$ is easily traceable. You should ensure that the angle measure is set to Radians in the **MODES** view if you intend to use this scale.

Retaining calculated values

When you find an extremum or an intersection, the point is remembered until you move the cursor even if it is not actually on a value that would normally be accessible for the scale you have chosen.



For example, if you find an intersection and then immediately return to the **FCN** menu and choose *Slope*, the slope calculated will be for the intersection just found rather than for the nearest pixel point. If you have recently found a root then pressing the **GOTO** key and entering the value **Root** will return the cursor to it.

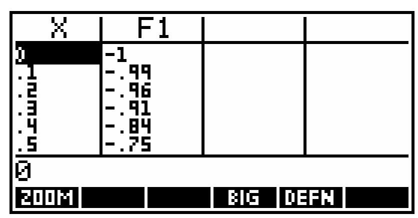
There are two ways to access these values:

- The first and simplest is via the value stored in memory **X**. If you move from the **PLOT** view to the **HOME** view without moving the cursor and type **X** then the value it will contain will be the last position of the cursor. If you just found a root or an intersection then this will be the value displayed. To find the y value for the x coordinate just evaluate **F1(X)** in the **HOME** view (or whatever function you are using).

- The second way is via the reserved words of **Root**, **Extremum**, **Area**, **Slope** and **Isect**. Typing any of these reserved words in any situation will retrieve their last calculated values. If a value has not yet been found then they will return zero.

This trick is particularly useful when calculating areas under or between curves. See page 75.

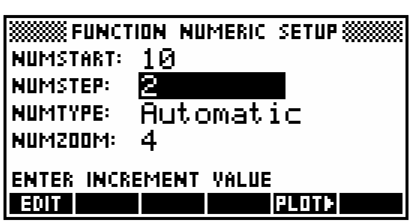
The NUM view revisited

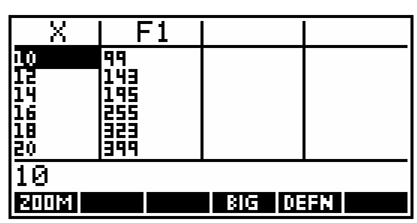


We saw earlier that the **NUM** view gives a tabular view of the function.

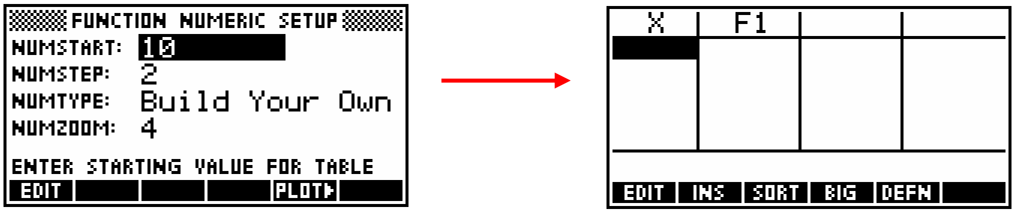
It is possible to manipulate this view through the **NUM SETUP** view. The first way is to change the start value and the step size for the view.

For example, values of 10 and 2 give:





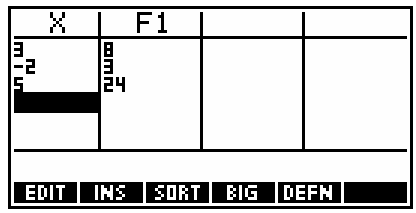
Looking at the **NUM SETUP** view you will see an entry called **NumType** with the default value of *Automatic*. The alternative to *Automatic* is the setting of *Build Your Own*. Under this setting the **NUM** view will be empty, waiting for you to enter your own values for **X**.



Typing in the values of (for example):

3 ENTER (-) 2 ENTER 5 ENTER

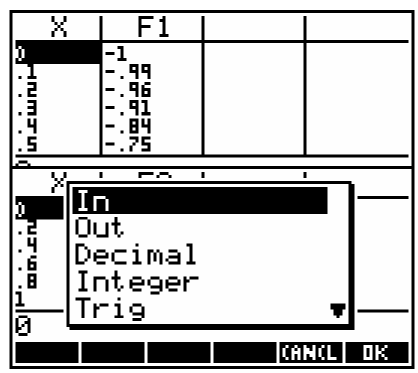
… will give…



In this situation the function values are being calculated as you input the **X** values. This can be quite useful if you are wanting to evaluate the behavior of a function at selected points.

If you now use the **NUM SETUP** view to switch to *Automatic* you will find that there is a **ZOOM** key at the bottom left of the **NUM** view. This can be quite useful as a fast way to reset the scale.

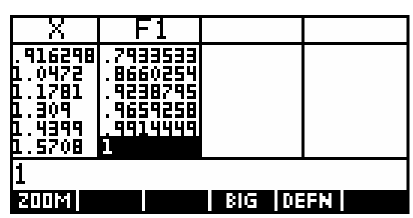
Pressing the **ZOOM** key pops up the menu on the right. The first option of *In* causes the step size to decrease from 0.1 to 0.025. This is a factor of 4 and is changeable via the **NUM SETUP** view. I find a zoom factor setting of 2 or 5 to be more useful.



The second option of *Out* causes the opposite effect, changing the step size upwards by whatever the Zoom Factor is set to.

The *Decimal* option restores the default settings. It changes from whatever is showing back to the step size of 0.1 The *Integer* option on the other hand, changes the scale so the step size is 1.

The *Trig* option changes the scale so that the step size is $\pi/24$ exactly. This will obviously be useful when dealing with trigonometric functions since it means that values such as $\pi/2$ will appear exactly in the table.



Integration: The definite integral using the $\int$ function

The situation for integration is very similar to that of differentiation. As with differentiation, the results for algebraic integration are better in the Function aplet. The $\int$ symbol is obtained via the keyboard.

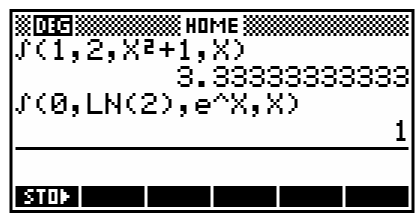
The syntax of the integration function is:

> $\int(a,b,\ function, name)$
>
> where: *a* and *b* are the limits of integration
> and *function* is defined in terms of *name*.
>
> eg. $\int(1,\ 3,\ x^2+5,\ x)$ or $\int(0.5,\ \pi,\ \cos(S1),\ S1)$

Let's look first at the definite integral…

The screen left shows $\int_1^2 x^2+1\ dx$ = 3.3333…

followed by $\int_0^{\ln 2} e^x\ dx$ = 1.



It may help you to remember the syntax of the differentiation and integration functions if you realize that they are filled in with values in exactly the same way that they are spoken.

E.g. $\int_1^2 x^2+1\ dx$ is read as:

"the integral from 1 to 2 of x^2+1 dx "

& entered the same way: $\int$ **(1, 2, X^2 + 1 , X)**

A similar path was taken with the differentiation function, so that:

$\frac{d}{dx}\left[X^2\right]$, which is read as "the derivative" ….. "with respect to X"….."of X^2"

and entered as it is read as $\partial X\left(X^2\right)$.

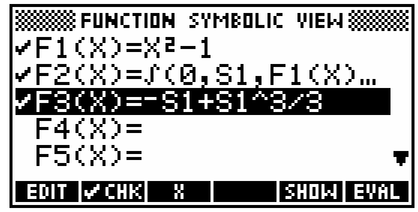
Note that although I have used **X** as the variable of integration in the examples above, this is not a requirement. The integration example could be done as $\int$ **(1, 2,** A^2+1 **, A)** with the same result.

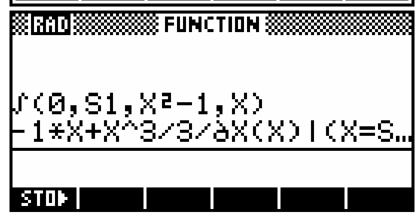
Integration: The algebraic indefinite integral

Algebraic integration is also possible (for simple functions), in the following fashions:

- If done in the **SYMB** view of the Function aplet, then the integration must be done using the symbolic variable **S1** (or **S2**, **S3**, **S4** or **S5**). If done in this manner then the results are satisfactory, except that there is no constant of integration 'c'.

  The screenshot right shows the results of defining $F1(X) = X^2 - 1$ and then $F2(X) = \int(0, S1, X^2 - 1, X)$, together with the results of the same thing after pressing the **EVAL** key (the result is placed in **F3(X)** only for convenience of viewing).

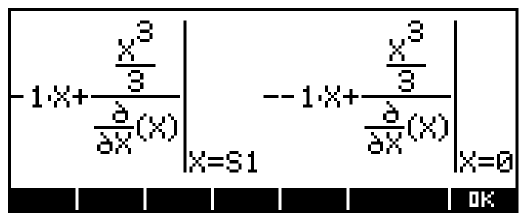




  All that is now necessary is to read '**-S1+S1^3/3**' as $-x+\frac{x^3}{3}$,

  or as it should be read as $\frac{x^3}{3} - x + c$.

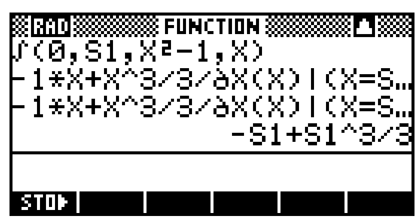
- If done in the **HOME** view, then **S1** must again be used as the variable of integration.



  i.e. $\int x^2 - 1\, dx$ is entered as

  $\int$ **(1, S1, X² - 1, X)**.

  This is shown above, together with the results of highlighting the answer and pressing **SHOW**. The result may seem odd but is caused by calculator assuming that **X** itself may be a function of some other variable and integrating accordingly as a 'partial integration'.
  While mathematically correct, this is not what most of us want.



  The way to simplify this answer to a better form is to highlight it, **COPY** it, and press **ENTER** again, giving the result shown right. This is the result of the calculator performing the substitutions implied in the previous expression.

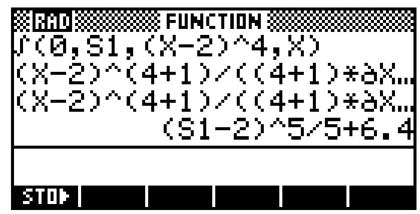
This substitution process has one implication which you need to be wary of and so it is worth examining the process in more detail. The expanded version of what is happening is shown below.

$$\int_0^{S1} x^2 - 1\,dx = \left[\frac{x^3}{3} - x\right]_{x=0}^{x=S1}$$
$$= \left(\frac{S1^3}{3} - S1\right) - \left(\frac{0^3}{3} - 0\right)$$
$$= \frac{S1^3}{3} - S1$$

The potential problem lies with the second line, where the substitution of zero results in the second bracket disappearing. This will not always happen.

For example…

$$\int_0^{S1} (x-2)^4\,dx = \left[\frac{(x-2)^5}{5}\right]_{x=0}^{x=S1}$$
$$= \left(\frac{(S1-2)^5}{5}\right) - \left(\frac{(0-2)^5}{5}\right)$$
$$= \frac{(S1-2)^5}{5} - 6\cdot 4$$



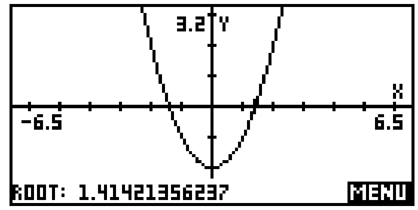
The final constant of 6.4 comes from substituting zero into the expression and should not be there if we were doing this with the aim of finding an indefinite integral. On the other hand, we all know that the answer should have a constant of integration, so perhaps this extra constant will help you to remember the '+c'!

Calculator Tip

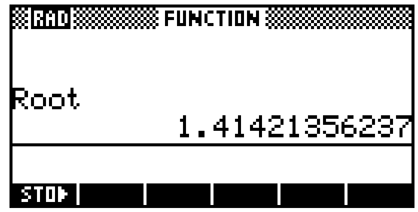
There are strict limits to what the hp 39gs can integrate. For example, on the hp 39gs if you try to evaluate $\int \sin^2 x.\cos x\,dx$ it will not be able to do it. Essentially, little beyond polynomials.

On the other hand, the hp 40gs is far more capable through its CAS (see page 324). The hp 40gs can handle integration by parts and by substitution as well as many other tricks.

Integration: The definite integral using PLOT variables

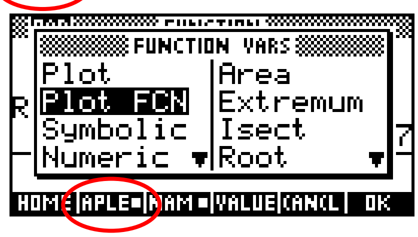


As was discussed earlier, when you find roots, intersections, extrema or signed areas in the **PLOT** view, the results are stored into variables for later use.



For example, if we use **FCN** *Root* to find the x intercept of $f(x)=x^2-2$ then the result is stored into a variable called **Root**, which can be accessed anywhere else. Similar variables called **Isect**, **Area**, and **Extremum** are stored for the other **FCN** tools.

You can access these names by typing them in using the **ALPHA** key, or by using the **VARS** key. Press **VARS** and you will see a list of the **HOME** variables.

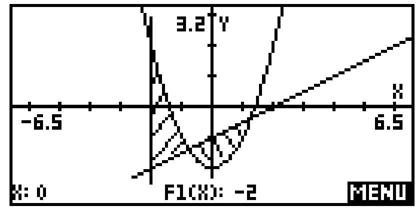


If you press **SK2**, labeled **APLET** (<u>not</u> the APLET button on the keyboard), then the display changes to show the variables specific to whatever aplet you are currently using. Those shown right are for the Function aplet and the group of *Plot FCN* variables is shown.

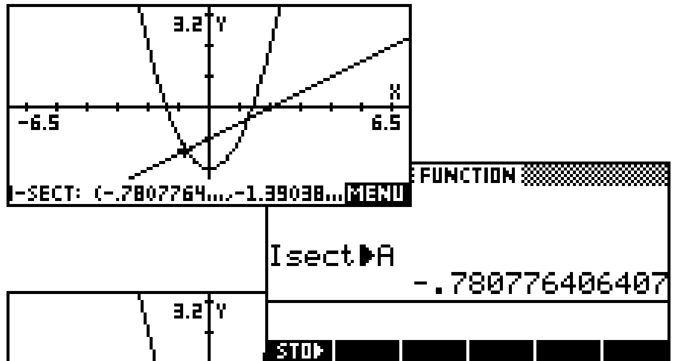
If you look at the screen shot you will see that the **NAME** tag is currently selected (showing as **NAM▪**) and this means that when you press **OK** the name of the variable will be pasted into what ever view you were using when you pressed the **VARS** key. Pressing the **VALUE** screen key will cause the *value* of the variable to be pasted instead. There is no effective difference in most situations.

This can be very useful in finding areas, as you will see in the example on the following page.

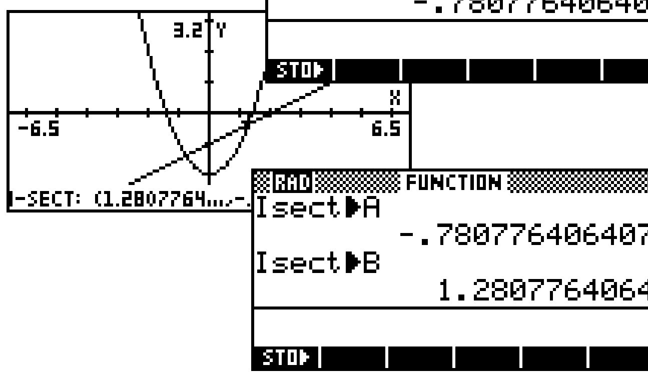
Suppose we want to find the area between $f(x) = x^2 - 2$ and $g(x) = 0.5x - 1$ from x = -2 to the first positive intersection of the two graphs.



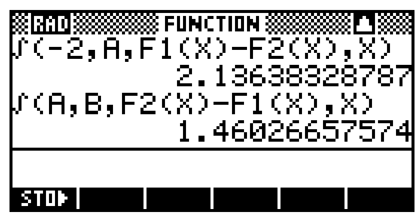
From the hand shaded screenshot shown above right it can be seen that to find the area we need to split it into two sections, with the boundaries being -2 and the two intersections.



After finding the first intersection using **FCN** *Intersection* we change into the **HOME** view and store the results into memory variable **A**.



We then do the same thing for the second intersection, storing the result into **B**.



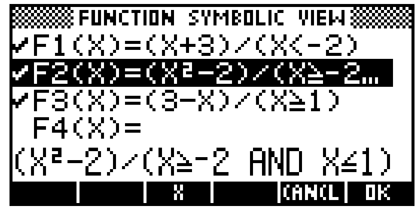
We can now calculate the area in the **HOME** view, using $f_1 - f_2$ for the first and $f_2 - f_1$ for the second. Use **COPY** to duplicate the first integral and edit it to adjust the functions and limits.

Finally, **COPY** the two solutions and add them to give the final answer.

Piecewise defined functions

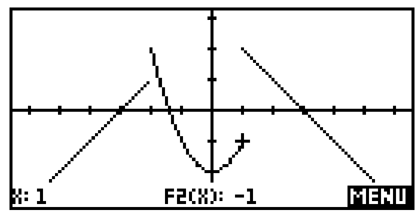
It is possible to graph piecewise defined functions using the Function aplet, although it involves literally splitting the function into pieces.

For example: $$f(x)=\begin{cases} x+3 & ;\ x<-2 \\ x^2-2 & ;\ -2\le x\le 1 \\ 3-x & ;\ x\ge 1 \end{cases}$$



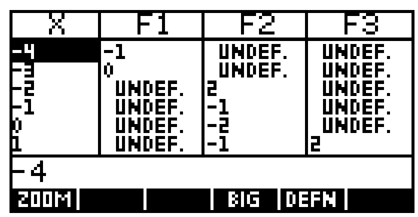
To graph this we need to enter it into the **SYMB** view as three separate functions:

F1(X)=(X+3)/(X < -2)
F2(X)=(X2-2)/(X ≥ -2 AND X ≤ 1)
F3(X)=(3-X)/(X ≥ 1)



Note: The **AND** function can be found on top of the **(-)** key.

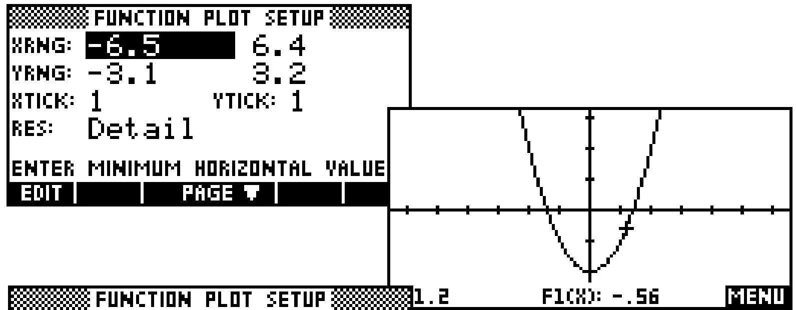
The reason why this works is that the **(X < -2)** and the **(X ≥-2 AND X ≤1)** expressions are evaluated as being either true (which for computers has a value of 1) or false (which has a value of 0).



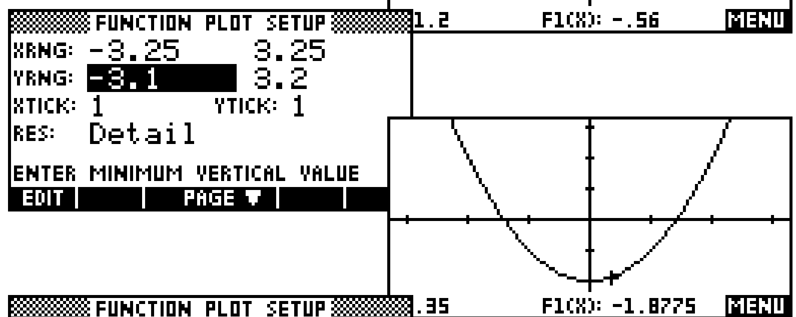
By dividing by this domain expression we are effectively dividing by 1 inside the range (with no effect) or dividing by zero outside the domain (making the function undefined). This can be seen in the **NUM** view to the right. Since undefined values are not graphed, this produces the desired effect.

'Nice' scales

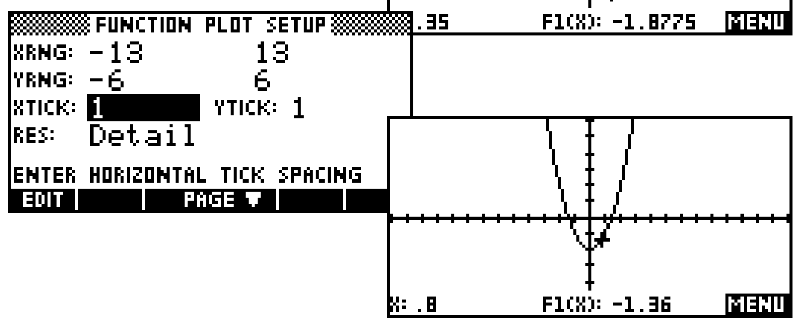
As discussed earlier, the reason for the seemingly strange default scale of -6.5 to 6.5 is to ensure that each dot on the screen is exactly 0.1 apart.



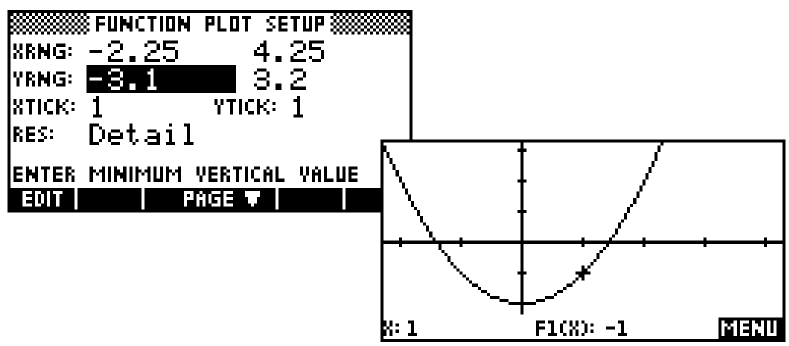
There are other scales, basically multiples of these numbers, that also give nice values if you want to **TRACE** along the graph. For example, halving each of -6.5 and 6.5 will place the dots 0.05 apart.



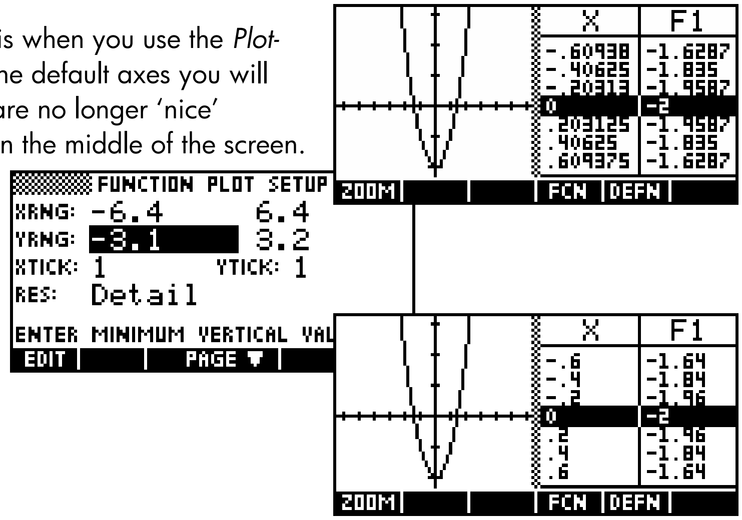
To zoom out instead of in simply double the values, producing dots that are 0.2 apart.



Similarly, if you want to center the graph around a particular value then just add that value to the range values. The example right is centered around x = 1 by adding 1 to -3.25 and 3.25.

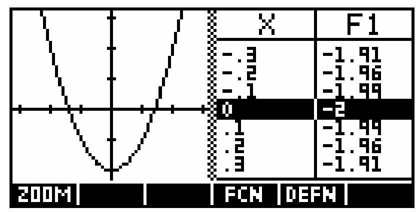


A time when 'nice' scales are more important is when you use the *Plot-Table* option in the **VIEWS** menu. If you use the default axes you will find that the dots, and hence the table values are no longer 'nice' because of the dots consumed by the line down the middle of the screen.



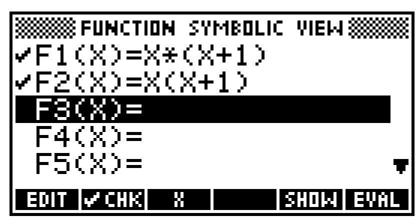
This can be solved by changing the x axis scale to -6.4 to 6.4, which gives table values of 0.2.

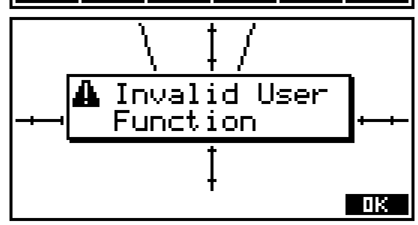
Using -3.2 to 3.2 is even better since it makes the graph 'square' again, with both axes proportional. Another good choice of scale for the *Plot-Table* view is -8 to 8, giving table values of 0.25. Basically any power of 2 is a good choice. Again, adding or subtracting a constant from each end of the axes will produce a graph where the y axis is not centred.



Use of brackets in functions

One problem commonly encountered by new users is misinterpretation of brackets. The hp calculator will correctly interpret **F1(X) = X^2(X+1)** as **X^2*(X+1)** but will not understand **F(X)=X(X+1)**. When used in either Function or Solve, it will result in the error message of "Invalid User Function".





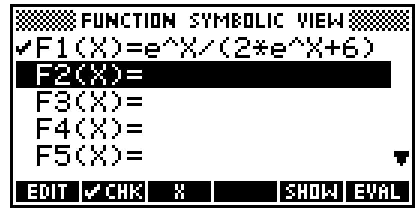
Similarly if you want to use the sum to n terms formula for a GP in the Solve aplet and enter it as **S=A(1-R^N)/(1-R)** then you will see a similar message unless you change it to read **S=A*(1-R^)/(1-R)**.

The reason for this apparent 'error' is that all of the built-in functions such as **SIN(....)** and **COS(....)** and **ROUND(....)** work with brackets. When the calculator encounters **X(X+1)** it interprets this as asking it to evaluate a function called **X(....)** at the value **X+1**. Since there is no such function it returns the error message that you are trying to use a function that is unknown.

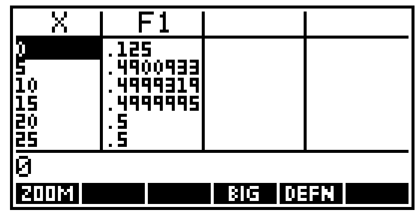
The solution is simple: just remember to put the ***** sign in when you use letters immediately before a bracket.

Problems when evaluating limits

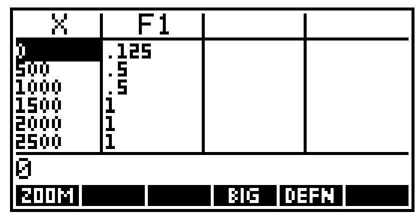
In evaluating limits to infinity using substitution, problems can be encountered if values are used which are too large.



For example: $\lim_{x \to \infty} \frac{e^x}{2e^x + 6}$



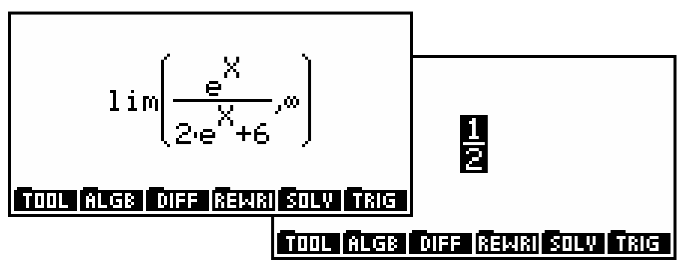
It is possible to gain an idea of the value of this limit by entering the function **F1(X)=e^X/(2*e^X+6)** into the Function aplet, changing to the **NUM** view and then trying increasingly large values. As you can see (right) the limit appears to be 0.5, which is correct. It is not the intention here to pretend that this is any sort of thorough mathematical justification but it does provide you with an indication of whether or not you are on the right track.



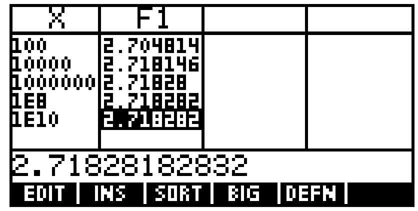
However, if you continue to use larger values then the limit appears to change to 1 (see right). This is obviously not correct, so why is it happening?

The reason for this is that the calculated value of e^x very quickly passes the upper limit of the capacity of the calculator, which is 10^{500}. When this happens the top and bottom of the fraction become equal (both at a value of 10^{500}) instead of the true situation of the bottom being roughly twice size of the top. This error is most likely to happen with limits involving power functions as they will overflow for smaller values of x.

The hp 40gs can instead evaluate limits algebraically using the CAS (see page 324). An example is shown right to illustrate the results.



A related effect happens when investigating the behavior of the commonly used calculus limit of $\lim_{n\to\infty}\left(1+\frac{1}{n}\right)^n$. One of the common tasks given to students in introductory calculus classes is to evaluate this expression for increasing values of *n* to see that it tends towards *e*. This can easily be done in the Function aplet using the **NUM** view but there is a trap in store for the unwary!

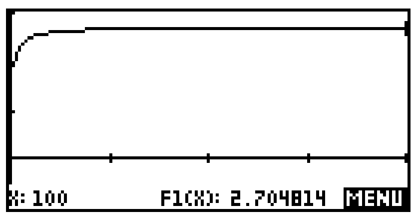


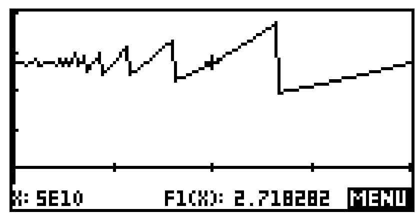
Begin as follows:

1. Entering the function into the **SYMB** view as **F1(X)=(1+1/X)^X**
2. Change to the **NUM SETUP** view and choose **Build Your Own** in the **NumType** field.
3. Now change to the **NUM** view enter increasingly large values for **X**.

The convergence towards e can also be seen graphically in the **PLOT** view but is very slow to reach high accuracy.

The 'trap' mentioned earlier lies in the fact that the slow convergence will mean that people will often try to graph this function for very large values of **X**. The first graph on the right shows the graph of this function for the domain of 0 to 100. The second graph shows how instability develops in the domain 0 to 1E11 (1×10^{11}).

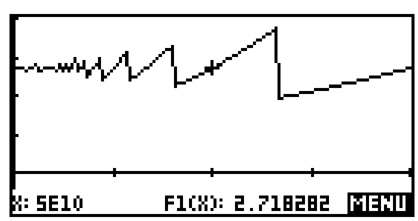




This apparent instability is caused by the internal rounding of the calculator. It works to 16 bits accuracy, which means that it can store 12 significant digits (for reasons only of interest to programmers). This means that when you invert a really large number and add it to one, you lose a lot of accuracy.

For example, if X = $2\cdot85\times10^{10}$ then $^1/_X$ is $2\cdot5087719298\times10^{-11}$. When you add 1 to this, the calculator is forced to discard all but the last decimal place because it can only store 12 significant digits. Thus 1 + $^1/_X$ becomes 1.00000000003 (rounded off from 1.00000000002508...)

There are naturally a whole range of numbers which will all round off to the same value of 1.00000000003, so that (for that range of numbers) the expression $(1+^1/_X)^X$ is equivalent mathematically (on the HP) to 1.00000000003^X. This produces a short section of an exponential graph, which only looks linear because you don't see enough of it.



Eventually the calculator reaches a value on the x axis which is large enough that it rounds off to a smaller number than 1.00000000003, which is 1.00000000002. This produces the sudden drop in the graph as the plot changes from a section of a 1.00000000003^X graph to a section of a 1.00000000002^X graph (which has a shallower gradient).

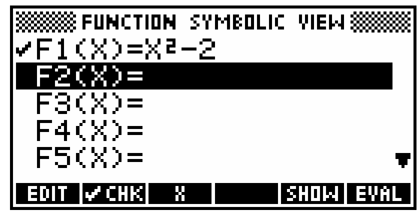
This section is maintained until the next drop, and so on. Finally, at the value $x = 2\times10^{11}$ the inverted value is so small that $1+{}^{1}/_{X}$ becomes exactly 1 and the graph becomes horizontal. Of course this is completely the wrong value!

Although this explanation may be beyond the level of many students it is quite important that they have some understanding of these ideas if they use an hp 39gs to numerically evaluate limits. Because of the CAS on the hp 40gs this situation is less likely to be a problem for that model. The solution to all problems of this type is to simply be aware of their existence and to allow for them rather than simply accepting the results shown in the **NUM** view.

Gradient at a point as the limit of the slope of a chord

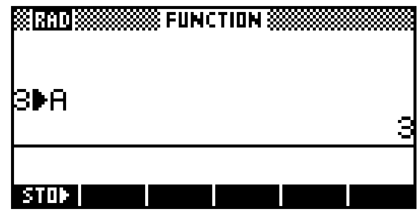
The true gradient at a point is available in a number of ways. For example, via the **FCN** *Slope* tool in the **PLOT** view or via the δ differentiation operator. For students first being introduced to calculus a common task is to investigate the slope of the chord joining two points as the length of the chord tends towards zero.

ie. $$\lim_{h\to 0}\left(\frac{f(x+h)-f(x)}{h}\right)$$

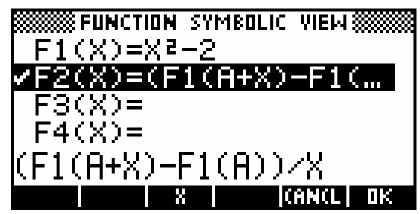


This can be effectively introduced via the Function aplet.

Begin by entering the function being studied into **F1(X)** as shown.



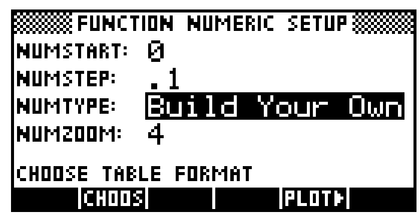
To examine the gradient at x=3, store 3 into memory **A** in the **HOME** view as shown right.



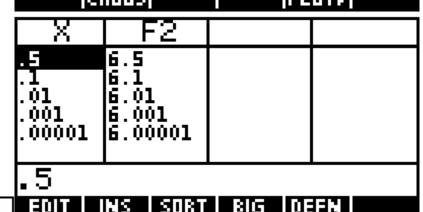
Return to the **SYMB** view, un-**✓CHK** the function **F1(X)** and enter the expression:

F2(X)=(F1(A+X)-F1(A))/X in **F2(X)**.

This is the basic differentiation formula quoted above with **X** taking the role of *h* and **A** being the point of evaluation, in this case with **A = 3**.

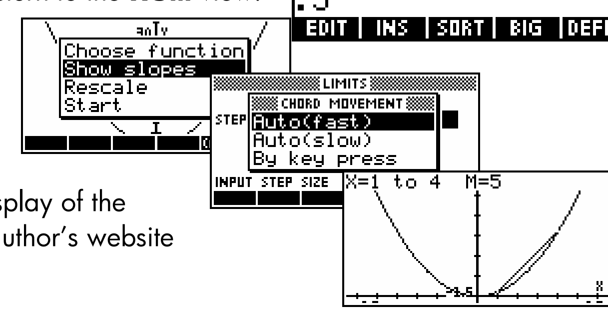


Change to the **NUM SETUP** view and change the **NumType** to "Build Your Own". By entering successively smaller values for **X** you can now investigate the limit as *h* tends towards zero.



In this case it is clear that the limit for x=3 is the value 6.

To investigate the gradient at a different point simply change back to the **HOME** view, enter a new value into **A** and then return to the **NUM** view.

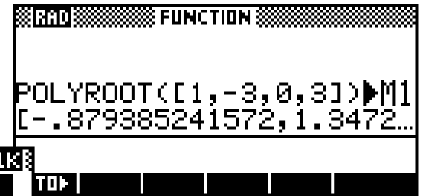


The disadvantage of the previous method is that it is not very visual. An alternative is to use an aplet downloaded from the web. An aplet that will automate the process and provide a visual display of the chord diminishing in length can be found on the author's website at http://www.hphomeview.com.

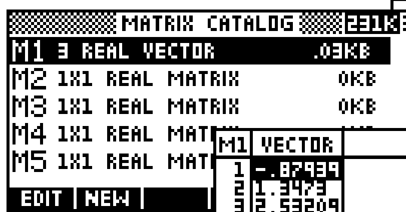
Finding and accessing polynomial roots

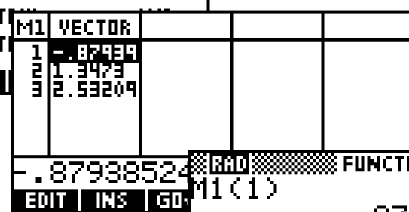
The **POLYROOT** function can be used to find roots very quickly, but the results are often difficult to see in the **HOME** view due to the number of decimal places spilling off the edge of the screen, particularly if they include complex roots. This can be dealt with easily by storing the results to a matrix.

For example, suppose we want to find the roots of $f(x) = x^3 - 3x^2 + 3$. We will use the **POLYROOT** function and store the results into **M1**.

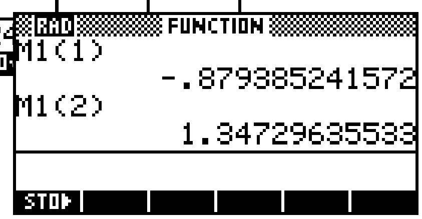


The advantage of this is that you can now view the roots by changing to the **Matrix Catalog** (**SHIFT 4**) and pressing **EDIT**. See page 209 for more detailed information on matrices.





In addition to this, you can access the roots in the **HOME** view as shown. This method works equally well for complex roots.



See page 309 for details on finding roots of real and complex polynomials using the CAS on the hp 40gs.

Calculator Tip
This trick is particularly helpful if you are working with complex roots. Not only does it make it easier to re-use them it makes it easier to tell at a glance which are real and which complex.

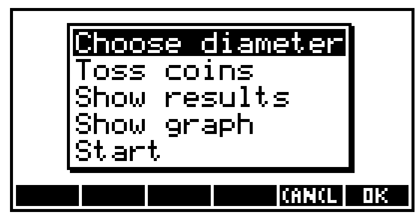
10

THE VIEWS MENU

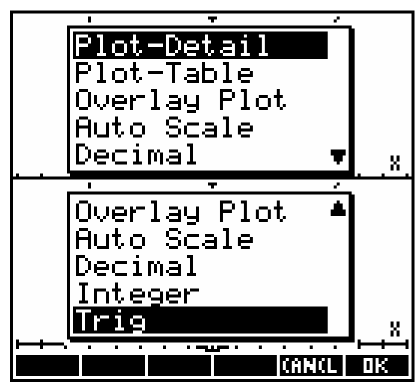
In addition to the views of **PLOT, SYMB** and **NUM** (together with their **SETUP** views), there is another key which we have so far only used very fleetingly - the **VIEWS** key.

It may seem odd to devote an entire chapter to what might appear to be an inconsequential key. In fact, however, this button is very useful to the effective use of the calculator, and crucial if you intend to use aplets downloaded from the internet.

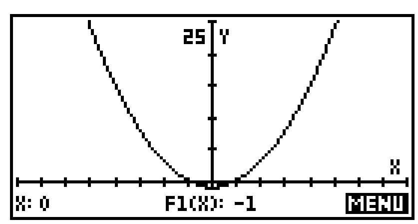
The contents of the **VIEWS** menu changes slightly according to which aplet you are currently using. The Function aplet contents are covered here but the others differ in only small ways.



Aplets downloaded from the Internet, however, will usually have a radically different **VIEWS** menus created by the person who wrote the program for the aplet. See page 257 for more information on this process if you intend to program the calculator. The first image shown right shows the contents of the **VIEWS** menu for an aplet called “Coin Toss” which investigates probability.



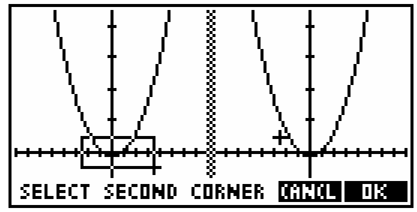
The **VIEWS** key normally pops up the menu on the right. We have already seen the use of the *Auto Scale* option (see page 89) and this is probably the one most used. The other options are also very useful at times.



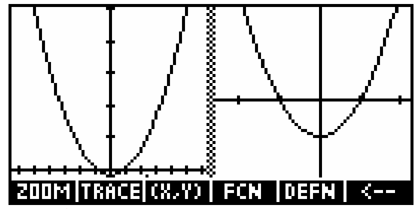
Shown on the right is the graph of $F1(X) = X^2 - 1$. If we press the **VIEWS** key, the menu above will pop up.

On the pages following we will investigate each of the menu options in detail.

Plot-Detail

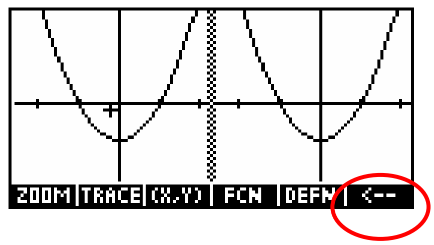


Choosing *Plot-Detail* from the menu splits the screen into two halves and re-plots the graph in each half. The right hand side can now be used to **ZOOM** without affecting the left screen. The idea is that you **ZOOM** on the left screen and the result appears on the right screen.

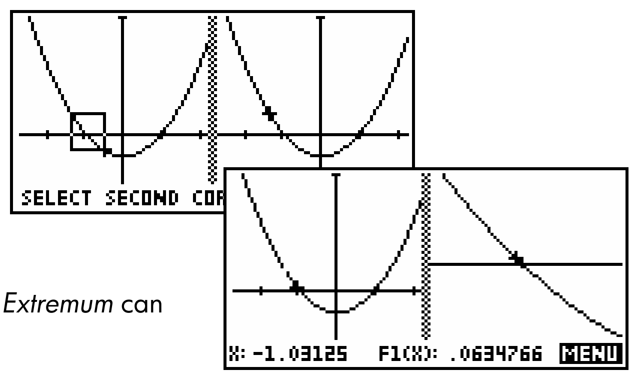


For example a *Box* zoom shows the result on the right allowing easy comparison of 'before' and 'after' views.

All the normal function tools are available except **GOTO**.



The left hand graph is always the active one, with results of actions shown on the right. We can now use the left graph again to zoom in on another section of interest, or alternatively, press the key under the **<--** label. This switches the right hand graph onto the left screen so that you can perform progressive zooms.

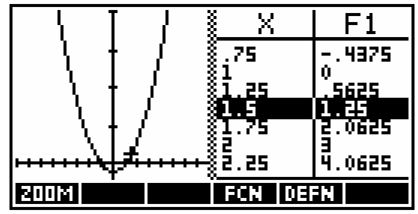


Using **TRACE** or the **FCN** menu you can then find or examine points of interest. Alternatively you can zoom in again using another *Box* zoom.

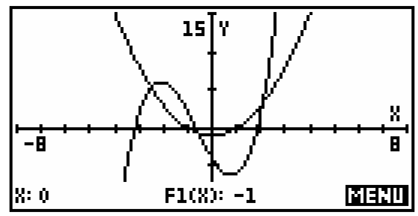
Any of the normal **FCN** tools such as *Signed Area* or *Extremum* can be used in this split screen.

Plot-Table

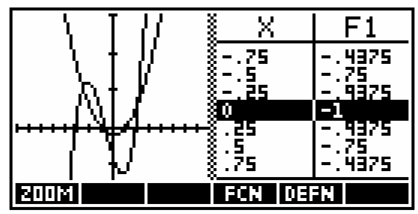
The next item on the **VIEWS** menu is *Plot-Table*. This option plots the graph on the right, with the Numeric view on the right half screen. Using the left/right arrow keys moves the cursor in both the graph and the numeric windows. See page 79 for information on how to keep nice scales in the table view. When more than one function defined in the **SYMB** view, pressing the up or down arrows changes the table focus from one to another. In this case, with only one, it centers the table.



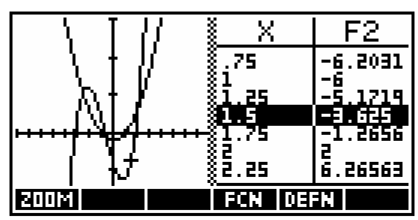
Let us switch now to a graph of the two functions $F1(X) = X^2 - 1$ and $F2(X) = X^3 + 2X^2 - 5X - 4$. This is shown on the right, using an **XRng** of -8 to 8.



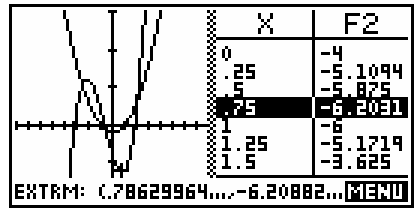
Choosing *Plot-Table* gives the result shown left. As you can see, the scale has been preserved unchanged, although without labels. The table on the right also uses a sensible scale of 0·25 because of the choice of an x axis scale of -8 to 8. As mentioned, choice of scale in the **Plot-Table** view is discussed in detail on page 79. The technique is requires slightly different values than the default values in the **PLOT SETUP** view.



Looking at the table heading you will see that it currently shows the function **F1(X)**. The left/right arrow keys move within that function, with the cursor keeping track. Pressing the up/down arrows now not only centers the table highlight but, more importantly switches from **F1(X)** to **F2(X)**. The **X** column does not change.



What makes this view even more useful is that the table keeps its 'nice' scale even while the usual **FCN** tools are being used. As you can see in the screenshot left, the table is automatically repositioned to show the closest pixel value to that of the extremum found.

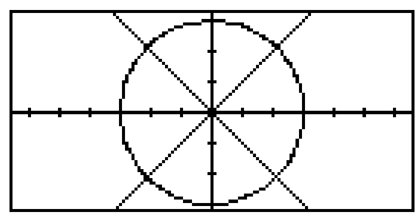


The *Signed Area* tool is also available in this view and when the cursor is moved the values in the table follow it. Unfortunately, the highlighted value in the table doesn't change as the cursor moves to create the shaded area. For this reason the best strategy is to use the **GOTO** key to jump to the end point. This means that the area will not be shaded but this should not be a problem.

Overlay Plot

Another possibility from the **VIEWS** menu is *Overlay Plot*. This option can be used to add another graph over the top of an existing one, without the screen being blanked first as it usually is. As an example, if you have already graphed functions **F1(X)** through to **F6(X)** and then add another one in the **SYMB** view, then you may not want to have to wait while all the earlier ones are redrawn. If you un-**✓CHK** the earlier graphs and then use *Overlay Plot* for the new one then it will be drawn over the top of the existing ones. Of course the results will not be good if the scales don't match!.

More usefully, you can use this to combine different styles of graphs. For example, the screen shot right was produced by drawing a circle in the Parametric aplet and then superimposing the equations y = x and y = -x in the Function aplet. This could be used to show, for example, the enclosing curves for conic sections.



You can also use this technique to overlay functions on top of statistical graphs. Again, the important point would be to ensure that the scales matched.

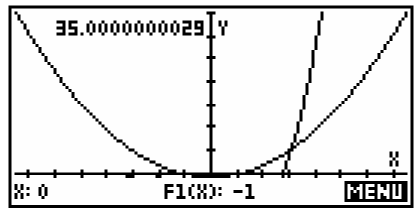
Auto Scale

Auto Scale is an good way to ensure that you get a reasonable picture of the graph if you are not sure in advance of the scale. After using *Auto Scale* you can then use the **PLOT SETUP** view to adjust the results.

It is important to understand two points about how *Auto Scale* works.

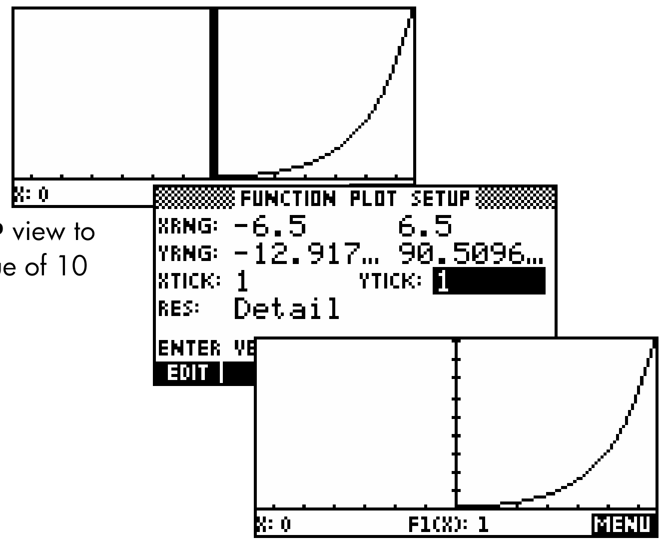
1. *Auto Scale* uses the X-axis range that is currently chosen in **PLOT SETUP**. It then adjusts the Y-axis range to include as much of the graph as possible. *It will not adjust the x axis.*

2. *Auto Scale* is done only for the ***first*** graph with a ✔CHK. If there are other graphs and they don't fit the scale then they will not benefit. As you can see in the example shown right, the quadratic shows well but the second graph (a cubic) shows only an ascending section. Zooming out would be an option at this stage, as would un-✔CHKing the quadratic in the hopes that *Auto Scaling* the cubic might give better results.



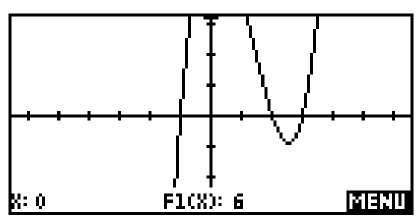
The resulting y scale is often not a very 'nice' one. Commonly you will find that the y axis appears 'thick' as shown right. The reason for this is that the **Ytick** value is too small, resulting in ticks too close together.

You will usually have to adjust it in the **PLOT SETUP** view to make it look good. The third graph has a **Ytick** value of 10 instead of 1.

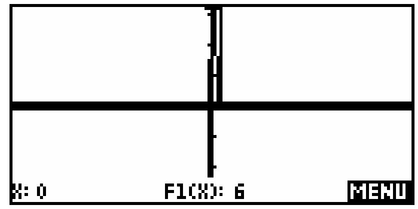


Decimal, Integer & Trig

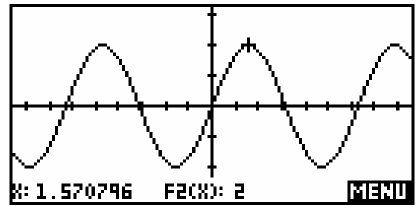
The next option of *Decimal* resets the scales so that each pixel (dot on the screen) is exactly 0·1. The result is an X scale of $-6\cdot5 \le x \le 6\cdot5$ and a Y scale of $-3\cdot1 \le y \le 3\cdot2$. This may not give the best view of the function. Personally I don't often use it, as it is generally easier to go to the **PLOT SETUP** view and press **SHIFT CLEAR**, which restores the factory settings to all fields.



The *Integer* option is similar to decimal, except that it sets the axes so that each pixel is 1 rather than 0.1 thus giving an X scale of $-65 \le X \le 65$. The usual result of this is rather horrible to be honest.



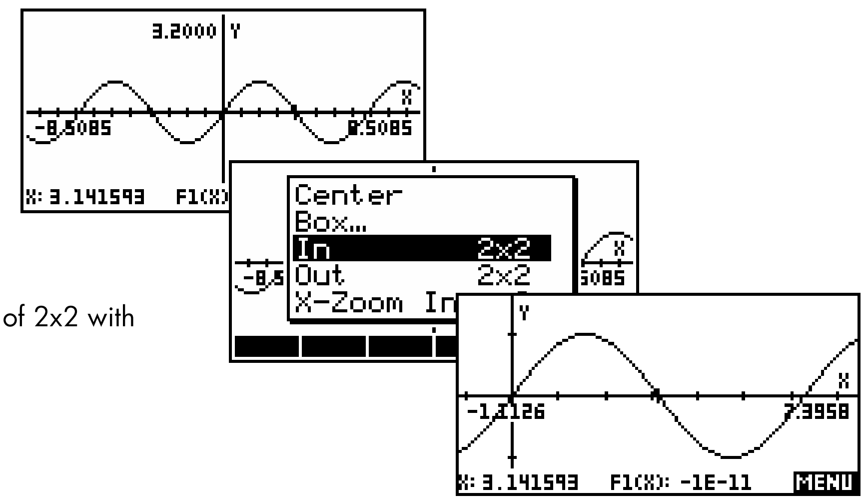
The final option of *Trig* is designed for graphing trig functions. It sets the scale so each pixel is $\pi/_{24}$. This means that if you were graphing **F1(X)=2SIN(X)** then pressing the left or right arrow 24 times would move you through exactly π and the value would be exactly 2 instead of a horrible decimal. This is shown right.



If you zoom in or out from this, the jumps will still stay relatively nice, particularly since 24 has so many factors. For example, with a zoom factor of 2, zooming out once would mean each pixel was now $\pi/_{48}$, while zooming in would give a pixel jump of $\pi/_{12}$.

The default axes under the *Trig* option is -2π to 2π. If you are primarily interested in the first 2π of the graph then simply change Xmin to zero. Alternatively you can move the cursor to π (the middle) and then zoom in.

The example right uses zoom factors of 2x2 with *Recenter:* ✓CHKed.



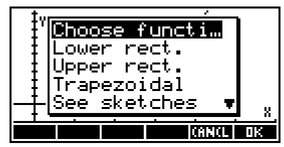
Calculator Tip

In the graphs above the cursor is at x = π. The coordinates at the bottom of the screen should show F1(X)=0 but doesn't due to the fact that the value of π stored internally is not exact and of course cannot be. The rounding of π in the 13th decimal place means that the resulting trig values will be 'wrong' in the 11th to 15th decimal place depending on the function used.

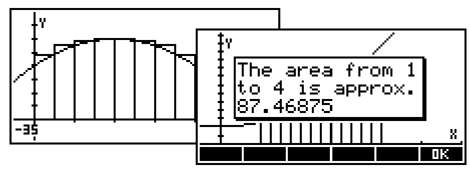
Downloaded Aplets from the Internet

The most powerful feature of the hp 39gs & hp 40gs is that you can download aplets and programs from the internet to help you to learn and to do mathematics. Two quick examples of aplets that are available are shown here. More are listed in the supplementary appendix on "Teaching Calculus using the hp 39gs & hp 40gs" and examples are also given in the chapter on programming. In each case the aplet is controlled by a menu. This menu is created by the programmer and 'attached' to the **VIEWS** button so that it displays in place of the normal menu.

Curve Areas

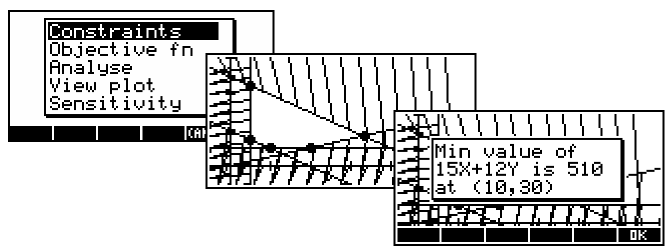


This aplet allows the user to find approximations to the area under a curve by finding either the lower rectangular area, the upper rectangular area, or the trapezoidal area.



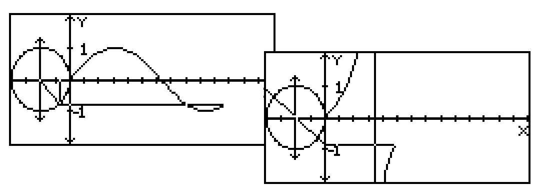
The user can choose the end points of the interval, the type of calculation and the number of rectangles to be used. The rectangles are drawn on the screen. A worksheet introduces the idea of integration to find areas.

Linear Programming



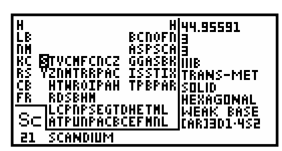
This aplet visually solves linear programming problems, finding the vertices of the feasible region and the max/min of an objective function. It even performs sensitivity analysis on the collected vertices.

Sine Define



This is an aplet to illustrate the process of obtaining the sine, cosine & tangent graphs from the unit circle.

Periodic Table



This is a library that can be added to the calculator which allows the user to display a periodic table with information on each element.

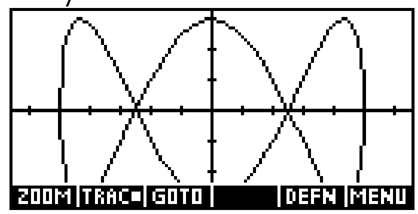
11

The Parametric Aplet

This aplet is used to graph functions where x and y are both functions of a third independent variable *T*. It is generally very similar to the Function aplet and so we will look mainly at the ways that it differs.

An example of a graph from this aplet is:

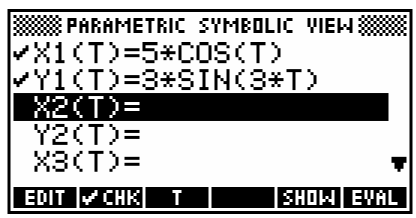
$$\left.\begin{aligned} x(t) &= 5\cos(t) \\ y(t) &= 3\sin(3t) \end{aligned}\right\} \; 0 \le t \le 2\pi \qquad \text{which gives:}$$



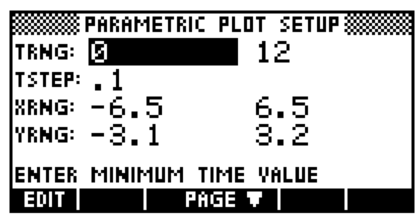
Although you can graph equations of this type, only some of the usual **PLOT** tools are present. As you can see in the screen shot above, the **FCN** key is not shown, meaning that none of its tools are available. Thinking about the nature of these equations will tell you why.

As usual the first step is to choose the aplet in the Aplet Library. Press the **APLET** key, highlight **Parametric** and press **START**. If you wish to ensure that you see the same thing as the examples following then press the **RESET** button before pressing **START**.

As with the Function aplet, this aplet begins in the **SYMB** view by allowing you to enter functions, but the functions are ***paired***. Each function consists of a function in T for X and another for Y.

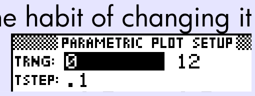


Looking at the **PLOT SETUP** view, you will see that we now have to enter a range for T as well as the usual ranges for X and Y. It is crucial to understand the different effect of the T range to that of the X and Y ranges.



Calculator Tips

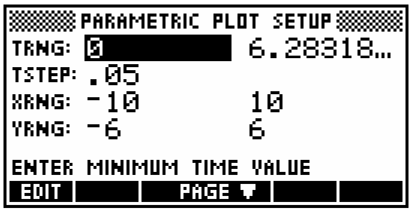
i. The default setting for TStep is 0.1. In my experience this is too large and can result in graphs that are not sufficiently smooth. It is worth developing the habit of changing it to 0.05



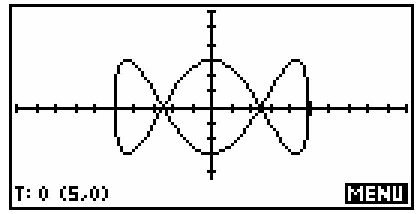
ii. The default maximum for T is 12. If your graph involves a trig function then this may not be a good choice. A better one might be 2π. You can use the pi above the [3] button to enter this.

The effect of TRng

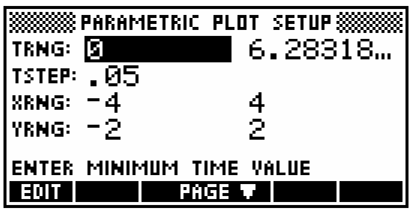
The X and Y ranges control the lengths of the axes. They determine how much of the function, when drawn, will be visible. See the examples below. Notice that in both cases, **TRAC** is on and shows the T value, followed by an ordered pair giving (X,Y).



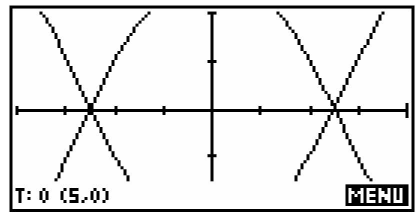
gives a graph of:



whereas..

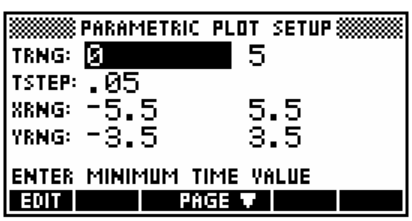


gives a graph of:

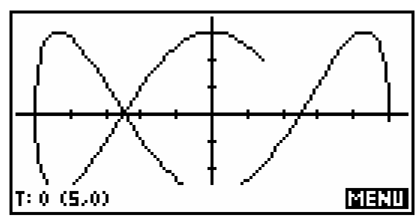


Unlike **XRng** & **YRng**, the effect of **TRng** is to decide how much of the graph is drawn at all, not how much is displayed of the total picture.

For example…

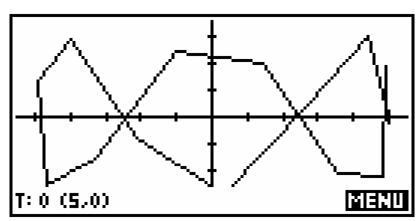


gives a graph of:



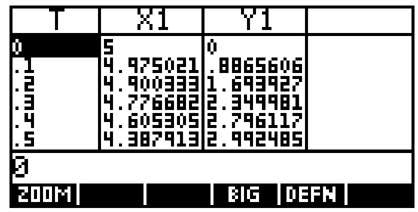
As you can see above, changing the T range from $0 \le t \le 2\pi$ to $0 \le t \le 5$ gives a graph that appears only partially drawn. Of course, what constitutes "fully drawn" depends on the function used.

The value of the parameter **TStep** controls the jump between successive values of **T** when evaluating the function for graphing. Making **TStep** too large will produces a graph which is not smooth. The example on the right shows **TStep=0.5** instead of **0.05**.



Calculator Tip

- Decreasing **TStep** beyond a certain point will slow down the graphing process without smoothing the graph any further. Using 0.05 is generally enough.
- Since trig functions are often used in parametric equations, one should always be careful that the angle measure chosen in **MODES** is correct. The default for all aplets is radian measure.



As usual, the **NUM** view gives a tabular view of the function. In this case there are three columns, since **X1** and **Y1** both derive from **T**.

As with the Function aplet, it is possible to change the starting point and step size of the table, and also to change it into a *Build Your Own* type of table (see page 70).

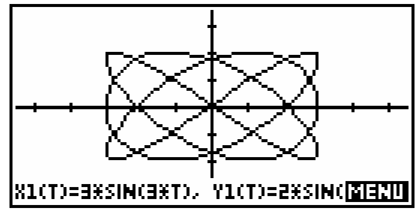
12

THE EXPERT: VECTOR FUNCTIONS

Fun and games

Apart from the normal mathematical and engineering applications of parametric equations, some interesting graphs are available through this aplet. Three quick examples are given below.

Example 1

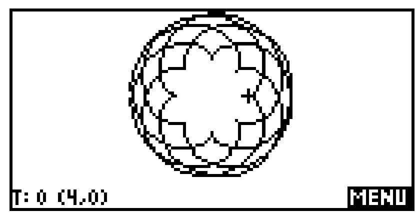


Try exploring variants of the graph of:

$$x(t) = 3\sin 3t$$

$$y(t) = 2\sin 4t$$

Example 2



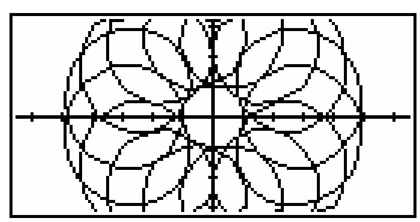
Try varying the values of *A* and *B* in the equations:

$$x1(t) = (A+B)\cos(t) - B\cos((A/B+1)t)$$

$$y1(t) = (A+B)\sin(t) - B\sin((A/B+1)t)$$

Hint: An easy way to vary A and B is to store values to memories **A** and **B** in the **HOME** view and enter the equations exactly as shown. New graphs can then be created by changing back to **HOME** and storing different values to **A** and **B**. The example shown uses **A=4**, **B=2.5** and has axes set with **TRng** of 0 to 31.5 step .2, **XRng** of -21.66 to 21.66 and **YRng** of -12 to 9. It also has **Axes** un-✓CHK'd in **PLOT SETUP.**

Example 2



Try varying the constants in the equations:

$$x1(t) = 3\sin(t) + 2\sin(15t)$$

$$y1(t) = 3\cos(t) + 2\cos(15t)$$

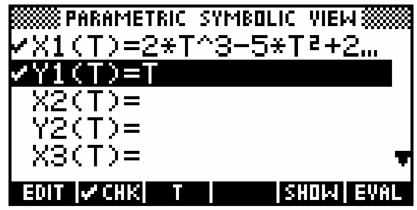
For those who remember them, this is curve like those produced by a "Spirograph".

Vectors

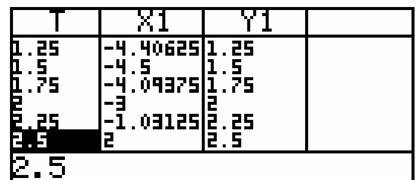
The Parametric aplet can be used to visually display vector motion in one and two dimensions.

Example 1

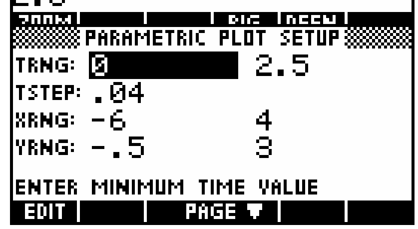
A particle P is moving in a straight line. Its velocity *v* (in ms^{-1}) at any time *t* (in seconds, *t>0*) is given by $v(t) = 2t^3 - 5t^2 + 2t - 3$. Illustrate its motion during the first 2.5 seconds.



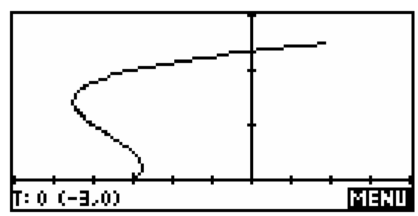
Enter the motion equation from (v) as X(T) and enter Y(T)=T. The only purpose of this second equation is to move the particle up the y axis as it traces out its path, thereby making it easier to view.



Changing to the **NUM** view lets me scroll through the first three seconds of movement, allowing me to choose a good scale for the x axis.



I'm interested in the first 2½ seconds only, so I'll also restrict TRng to 0 to 2.5. Using Y(T)=T for this TRng means the y values will also range from 0 to 2.5. Maximizing visibility of this range of values is the reason for setting YRng to be –0.5 to 3 in **PLOT SETUP**. The range for the x axis is chosen from the values shown in the **NUM** view . The value of TStep is carefully chosen so that when the motion plots, the speed is slow enough to show its progress.



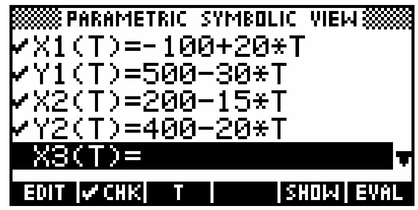
The graph makes it plain that it doubles back twice in the first two seconds. The really nice part about this method though is that the motion can be *seen* on the screen. As the particle slows down near the turning points it does so on the screen. As it accelerates in the final section you can see this on the screen too. Obviously this can't be seen on this page, and I recommend you try this for yourself.

Example 2

Two ships are traveling according to the vector motions given below, where time is in hours and distance in kilometers. Illustrate their motion during the first ten hours.

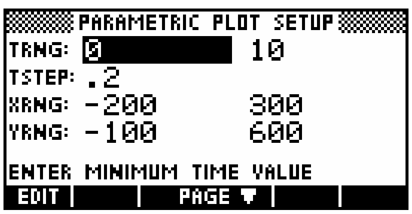
$$Ship\,A: \qquad \tilde{x}_A = \begin{pmatrix} -100 \\ 500 \end{pmatrix} + \begin{pmatrix} 20 \\ -30 \end{pmatrix} t$$

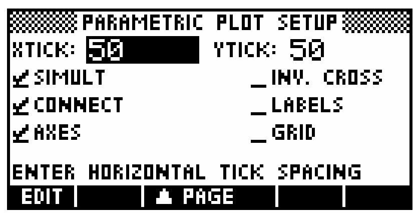
$$Ship\,B: \qquad \tilde{x}_B = \begin{pmatrix} 200 \\ 400 \end{pmatrix} + \begin{pmatrix} -15 \\ -20 \end{pmatrix} t$$



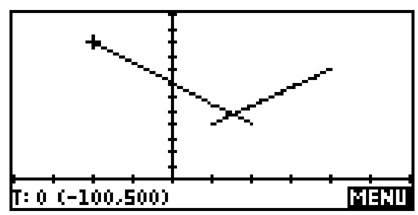
Enter the equations of motion as shown right.

Now change to the **PLOT SETUP** view and set the axes to suitable values. Possible values are shown below.

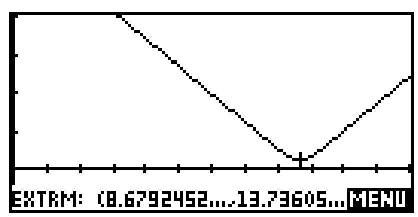




Now press **PLOT** to see the ships paths appear. As with the previous example, the value of **TStep** is chosen to allow visible motion. As the graph appears it can be seen that the ships do not collide, even though the final plot may make it appear that they do.



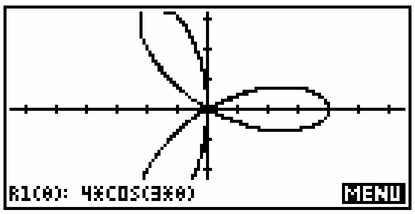
To find the distance between the two ships at any time *t*, you can enter the equation **F1(X)=** $\sqrt{\left(X1(X)-X2(X)\right)^2+\left(Y1(X)-Y2(X)\right)^2}$ into the Function aplet. Note that the active variable must be an **X** in the Function aplet instead of the **T** of the Parametric aplet but you can still refer to the Parametric functions **X1** and **Y1** even within the Function aplet. Graphing this function in the **PLOT** view of the Function aplet will allow you to find its minimum value. In this case the minimum separation of 13.74 km is achieved at *t=8.68* hrs.



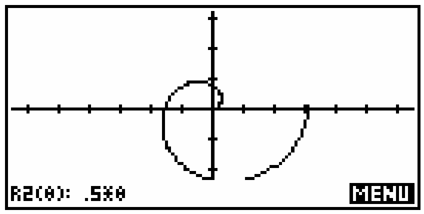
THE POLAR APLET

This aplet is used to graph functions of the type where the radius r is a function of the angle θ (theta). As with the Parametric aplet, it is very similar to the Function aplet and so the space devoted to it here is limited mainly to the way it differs. Some examples of functions of this type, together with their graphs are:

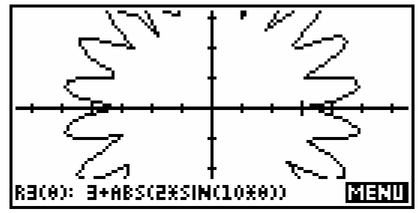
$R1(\theta) = 4\cos(3\theta)$



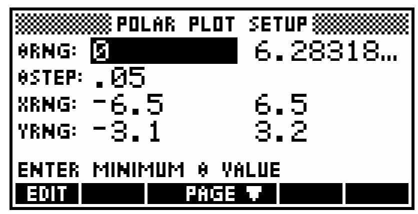
$R2(\theta) = 0.5\theta$



$R3(\theta) = 3 + |2\sin(10\theta)|$



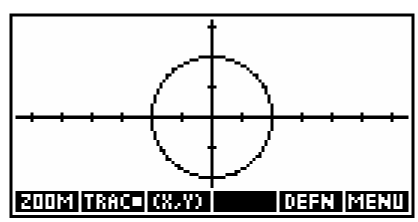
The **PLOT SETUP** view shows that ranges must be specified not only for **X** and **Y** but for θ. The values set for **XRng** and **YRng** control the length of the axes. The movement of the cursor follows the increments set for θ in **θStep** regardless of the resulting **X** and **Y** positions.



The values of **θRng** and for **θStep** are critical in controlling the appearance of the graph. The values set for **θRng** control how much of the graph is drawn, while the values for **XRng** and **YRng** only control how much of the graph is displayed on the screen once drawn. **θStep** controls how smooth the graph is, as did **TStep** in the Parametric aplet. The default values for **θRng** are $0 \le \theta \le 2\pi$ and the default value for **θStep** is $^{\pi}/_{24}$.

The default for **θStep** often results in graphs which are not very smooth and 0.05 is a better compromise between smoothness & speed. If you particularly want to use a fraction of π to aid in tracing the graph then try $^{\pi}/_{60}$.

If it is important that graphs of circles do not look oval, then use the default values for the axes. If the resulting axes don't show enough of the function, then **ZOOM** in or out. The **ZOOM** *Square* option can also be used to achieve round circles.



14

THE SEQUENCE APLET

This aplet is used to deal with sequences, and indirectly series, in both non-recursive form (where T_n is a function of n) and implicit/recursive/iterative form (where T_n is a function of T_{n-1}).

Examples of these types of sequences are:

(explicit/non-recursive)

$T_n = 3n - 1$ ..... $\{2, 5, 8, 11, 14, ...\}$

$T_n = n^2$ ..... $\{1, 4, 9, 16, 25, ...\}$

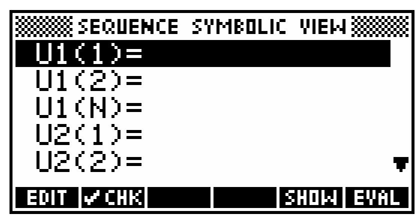
$T_n = 2^n$ ..... $\{2, 4, 8, 16, 32, ...\}$

(implicit/recursive)

$T_n = 2T_{n-1} - 1 \ ; T_1 = 2$ ..... $\{2, 3, 5, 9, 17, ...\}$

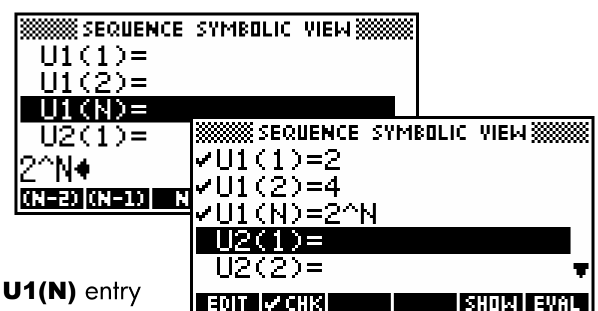
$T_n = 5 - T_{n-1} \ ; T_1 = 2$ ..... $\{2, 3, 2, 3, 2, ...\}$

$T_n = T_{n-1} + T_{n-2} ; T_1 = 1, T_2 = 1$ ..... $\{1, 1, 2, 3, 5, 8.....\}$



As with most aplets, the Sequence aplet starts in the **SYMB** view when you enter formulas. The Sequence aplet uses the terminology U(N) rather than the other commonly used T_n for its definitions in order to avoid having to use subscripts which would not show up well on the screen.

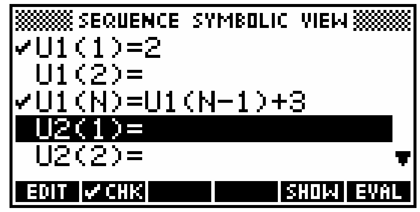
All functions of this type are assumed to be defined for the positive integers only – ie. for **N = 1,2,3,4...**



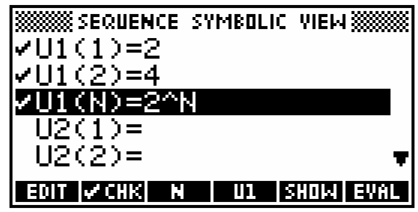
Each definition has three entries - **U1(1), U1(2)** and **U1(N)** (see above) but it is not always necessary to supply all three.

For example, if the sequence is non-recursive then only the **U1(N)** entry needs to be filled in, with the other two entries calculated automatically from the definition as shown in the sequence of two screens shown right.

If the definition is recursive but only involves T_{n-1} rather than both T_{n-1} and T_{n-2} then you need not enter a value for **U1(2)**.

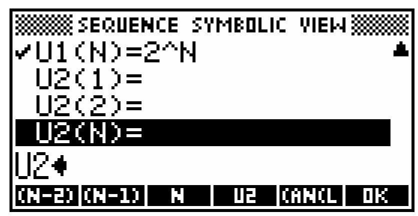


For example, for the sequence $T_n = T_{n-1} + 3; \quad T_1 = 2$ you need only enter the value **5** into **U1(1)** and the expression **U1(N-1)+4** into **U1(N)**. The value of **U1(2)** will be ignored in the **SYMB** view but filled in by the calculator automatically in the **NUM** view.



There are a number of very convenient extra buttons provided at the bottom of the screen when entering sequences.

Two of these - **U1** and **N** - are available as soon as the cursor moves onto the **U(N)** line (see right). Pressing either will enter the appropriate text into the sequence definition.

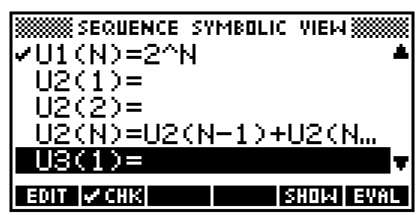


The rest become visible once you have begun to enter the sequence definition.

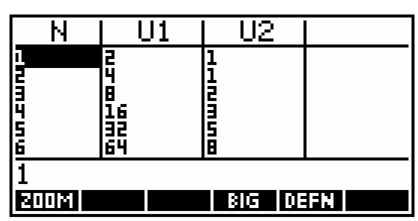
For example, suppose we enter the Fibonacci sequence into **U2** by defining **U2(N)** as **U2(N-1) + U2(N-2)**. Rather than having to type all of this we can use the buttons provided, pressing:

U2 **(N-1)** + **U2** **(N-2)**

This is a very convenient feature, and worth remembering.

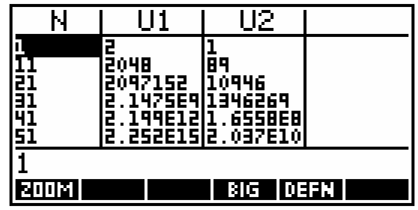


There is no **✓CHK** mark next to the definition yet, since the sequence is defined recursively and no values have yet been given for **U2(1)** and **U2(2)**. Type in a value of **1** for both of these and then press the **NUM** button to switch to the **NUM** view.

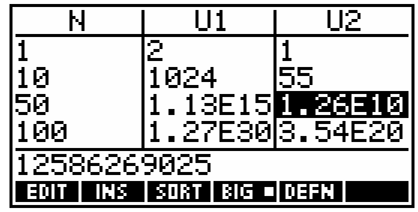


As you can see in the screenshot right, the **NUM** view shows the actual values in the sequence as a table. If you move the highlight into the **U1** and **U2** columns, you can press the **DEFN** button and see the sequence rule. You can experiment for yourself and see the result of pressing the **BIG** button (see next page for an example). The **ZOOM** button is also available as usual, but it is easier to use other methods.

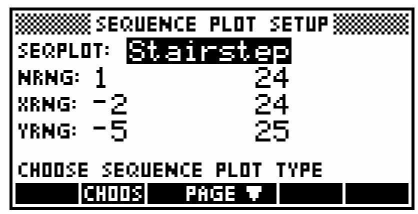
The **NUM SETUP** view offers more useful features. Change to that view now and change the **NumStep** value to 10. If you then swap back to the **NUM** view you will see (as right) that the sequence jumps in steps of 10. In case you don't realize… **2.1475E9** is 'computer speak' for $2{\cdot}1475\times10^{9}$.



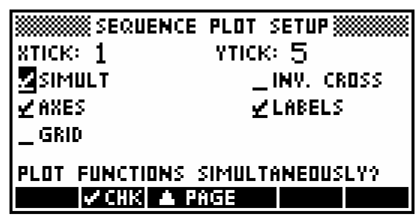
Now go back to the **NUM SETUP** view and change the *Automatic* setting to *Build Your Own* by moving the highlight to it and pressing the **+** key or by using **CHOOS**. Switch back to the **NUM** view and enter the values **1**, **10**, **50** and **100** into the **N** column.



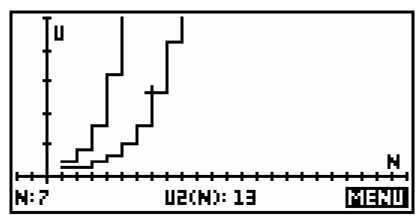
You will find that the values for those terms of each sequence will appear in the **U1** and **U2** columns almost immediately. In case you didn't realize, the reason for the larger text is that the **BIG** button has been pressed.



Due to the type of problems one is usually trying to solve with sequences, the **NUM** view rather than the **PLOT** view is often more useful in this aplet, but let's have a look at the **PLOT** view anyway. Two types of plots are available, the default being *Stairstep*.



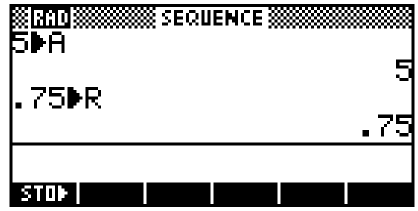
Change to the **PLOT SETUP** view and ensure that the setup view is the same as that shown in the two screens above right. Then change to the **PLOT** view and you should see a graph similar to the one shown below right. The second type of graph is the *Cobweb*.



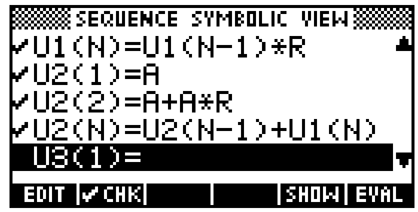
THE EXPERT: SEQUENCES & SERIES

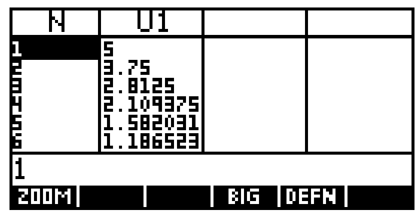
Defining a generalized GP and the sum to n terms for it.

If we define our GP using memory variables then it becomes far more flexible.



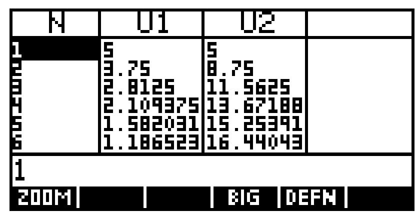
The advantage of this method is that you now need only change the values of **A** and **R** in the **HOME** view to change the sequence.





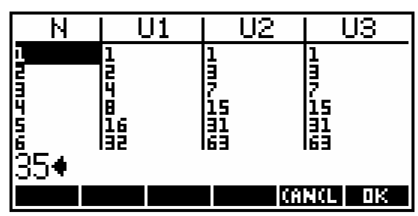
Defining a series (sum to n terms of a sequence) is fairly straight-forward using a similar method. Note the reference to **U1** in the definition of **U2**.

Once **U2** is defined in this way you can change both **U1** & **U2**. by simply storing new values into **A** and **R** from the **HOME** view.



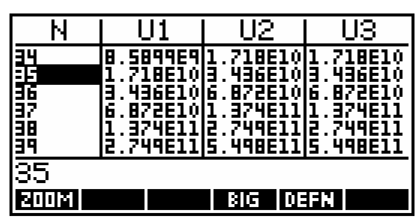
Solving sequence problems

Questions like "What term is the first to be greater than 10 000?" or "When does S_n first exceed 10 000?" can be answered in the **NUM** view.



Simply move into the **N** column, make a guess as to the term you require and type it in. The table jumps to that value.

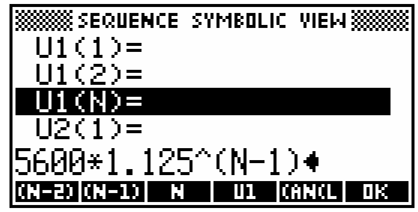
For example, in answer to the first question, we might estimate **N=35**. However we would find that **U1(35)** is far too large. By successive guesses, we find that T_{15} is the one we were seeking. The second problem is as easily solved in this way.



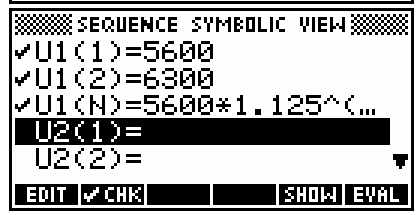
Population type problems are also easily dealt with in this way.

> For example, "A population of mice numbers 5600 and is growing at a rate of 12.5% per month. How long will it be until it numbers more than one million?"

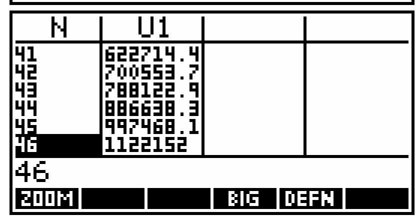
Pressing **CLEAR** (above **DEL**) clears out the existing expressions, and we can enter the formula for the GP modeling the situation shown right.



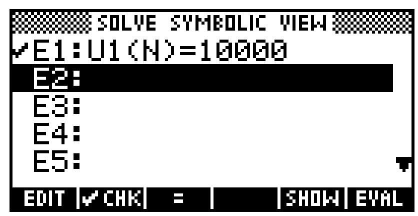
Because this is a non-recursive rule, the two initial values of 5600 and 6300 will be automatically calculated when you enter the rule into **U1(N)**.



All we need do now is switch to the Numeric view to find, with some experimenting, that **U1(46)** is the first to exceed one million.



It is also possible to answer these questions in the Solve aplet. For example if we use the Sequence aplet to define **U1(N)=2^(N-1)** as before, then we can change to the Solve aplet and enter into **E1** the equation **U1(N)=10000**, change to the **NUM** view and press **SOLVE** and obtain an answer of 14·29 . This means, of course that **N=15** is the first term exceeding 10000.

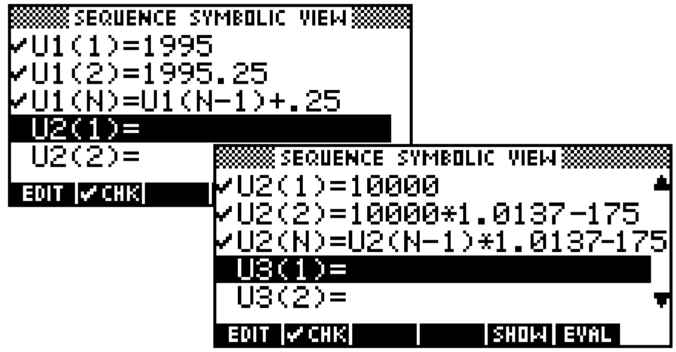


The reason why we did not use **E1(N)>10000** is that the Solve aplet is unable to deal with inequalities even though no error message is given if you try.

Modeling loans

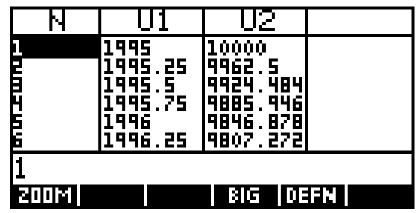
Suppose that I need to see the progress of a loan of $10,000 at a compound interest of 5.5% p.a. calculated each quarter, starting Jan. 1 1995, with a quarterly repayment rate of $175.

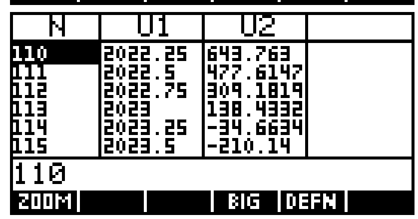
This problem can be modeled by a sequence.



To do this, set up **U1** and **U2** as shown above. The first sequence **U1** simply models the year, showing the quarters as .25, .5 and .75. The second sequence **U2** models the loan itself. Each quarter the remaining balance is multiplied by 1 + 0.055/4 and then the repayment of $175 is subtracted.

You can now follow the progress of the loan in the **NUM** view, with **U1** containing time and **U2** the amount owing at the start of each time period, showing it is repaid during the first quarter of 2023.





In reality this type of problem is more easily handled in the Finance aplet (see page 155). The Finance aplet has abilities that are specifically aimed at financial problems and allows you to solve for all parameters rather than simply modeling the problem as was done above.

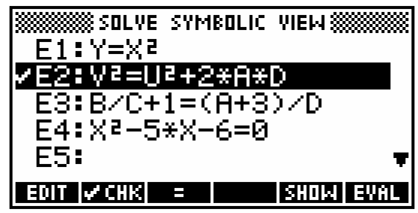
THE SOLVE APLET

This aplet will probably rival the Function aplet as your 'most used' tool. It solves equations, finds zeros of expressions involving multiple variables, and even involving derivatives and integrals.

To ensure that we are using the same terminology, let's define our terms first.

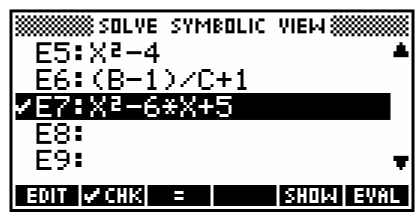
An equation includes an = sign, and can usually be solved:

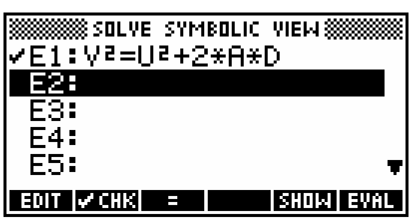
eg. $$\left.\begin{array}{l} y = x^2 \\ v^2 = u^2 + 2ad \\ \dfrac{b}{c} + 1 = \dfrac{a+3}{d} \\ x^2 - 6x + 5 = 0 \end{array}\right\} \text{...are all equations.}$$



An expression, on the other hand, does not contain an = sign. It can be evaluated or rearranged but not solved. When you enter an expression into the Solve aplet it internally puts an " = 0 " onto the end so as to convert it into an equation which can be solved.

eg. $$\left.\begin{array}{l} x^2 - 4 \\ \dfrac{b-1}{c} + 1 \\ x^2 - 6x + 5 \end{array}\right\} \text{... are all expressions.}$$





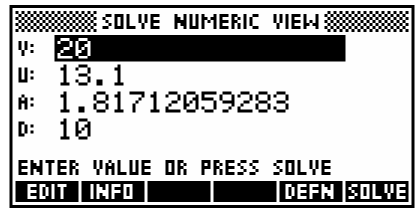
Let's start by looking at the equation $v^2 = u^2 + 2ad$. This equation gives the final velocity (*v*) of a object as a function of the initial velocity (*u*), the acceleration acting on it (*a*) and the distance traveled (*d*).

Suppose you had the problem:

> *"What acceleration is needed to increase the speed of a car from* 16·67 *m/s (60kph or ~38mph) to* 27·78 *m/s (100kph or 60mph) in a distance of 100m (~110 yd)?"*

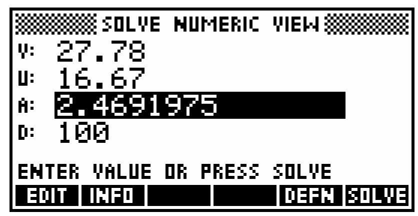
We'll assume that you have already entered the equation into **E2** (as above) and have made sure that it is **✔CHK**ed.

If you press **NUM** to change to the **NUM** view, you will see something similar to the screen on the right. What values are showing on your screen will depend on what happens to be in the memories **V**, **U**, **A** and **D** at the time. Pressing **SHIFT CLEAR** will zero all values.

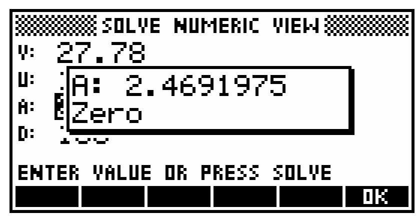


Move the highlight to **V** and enter the value 27·78, then to **U** and enter 16·67 and finally to **D** and enter 100.

Now move the highlight back to **A** (the value you're trying to find) and press the **SOLVE** button. You should find that you obtain the answer to our problem of 2·47 m/s^2.



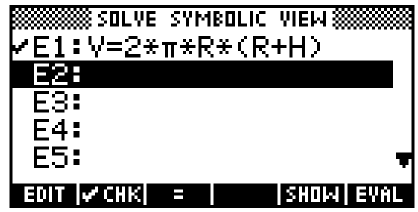
When the **SOLVE** process has finished, you can obtain a report on it by pressing the **INFO** button. The result in this case may not seem very informative but there is more about these messages on page 106. The information they supply can be critical and you should develop the habit of checking them.



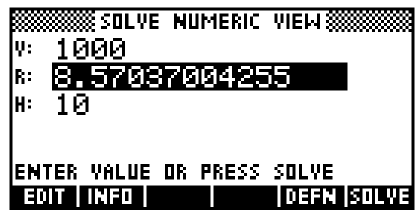
Our first example was fairly simple because there was only one solution so it did not much matter where we began looking for it. When there is more than one possible answer you are required to supply an initial estimate or guess. The Solve aplet will then try to find a solution which is 'near' to the estimate.

Example 1

The volume of a cylinder is given by $V = 2\pi r(r+h)$. *Find the radius of a cylinder which has a volume of 1 liter and a height of 10cm.*



Enter the equation into **E1** as shown right. When you are entering the equation, ensure that you put a * sign between the **R** and the bracket. See page 79 for more information on the reason for this.

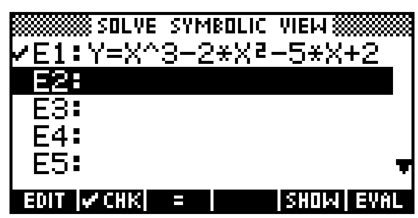


Change to the **NUM** view and enter the known values, remembering that 1 liter=1000cm^3. Position the highlight over **R**, enter a positive value as your estimate, and press **SOLVE** to find the solution shown right of 8.57cm. The equation is a quadratic in **R** which means two solutions are possible. If you enter an initial estimate of -10 you will obtain the negative solution, which is physically invalid in this case.

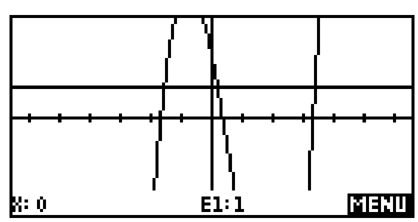
Example 2

If $f(x) = x^3 - 2x^2 - 5x + 2$ find all values of x for which $f(x) = 1$.

Although you may have a clear picture in your mind and can provide Solve with the estimates it needs, I'll assume that, like me, you would find it helpful to see a graph first.



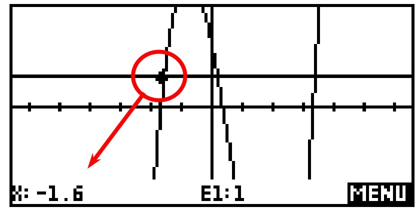
It is also possible to solve this in the Function aplet, which offers more powerful tools. The **PLOT** view in the Solve aplet, although powerful, can be deceptive if you don't understand it and I sometimes find it easier to work in the Function aplet. In this case we will continue to work in Solve.

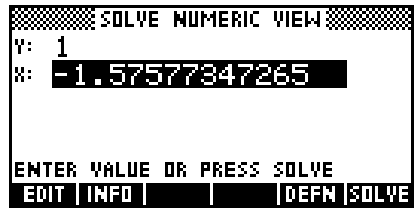


In the **SYMB** view, enter the equation **Y=X^3-2X^2-5X+2** into **E1**. In the **NUM** view, enter the known value of **Y=1**, ensure that the highlight is on **X**, making it the active variable, and then press **PLOT**. If your view does not look like this then you may not have had the highlight on the **X**, or your axes may not be set the same as mine in **PLOT SETUP**.

The **PLOT** view on the previous page shows two curves. The horizontal line is the left side of the equation which, when the known value of **Y=1** is substituted, forms a constant straight line. The other curve is the right hand side of the equation which, since **X** is the active variable, forms a cubic. In this case the scale chosen is appropriate but this will not always be the case and some adjustment might be required in the **PLOT SETUP** view.

We require values where the two curves intersect. Using the arrow keys, move the cursor near to the first intersection point. I found (see right) that -1.6 seemed to be a good approximation. Now change back to the **NUM** view and you will find that this approximation has been carried back as the initial estimate. Press **SOLVE** to find the true value.



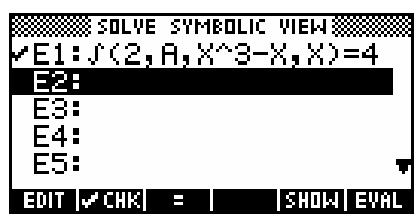


Repeat the process of obtaining an estimate in the **PLOT** view and refining it in the **NUM** view to find the other two solutions. See page 110 for more information on the effect of the active variable on what you see in the **PLOT** view.

Calculator Tip
The Solve aplet is not able to cope with inequalities. Although there is no error message when you use < or >, the answer it supplies is not what you would expect. What is worse is that they are reported in **INFO** as correct! Just don't use inequalities. If you have an hp 40gs then you can use the CAS to solve inequalities.

The Solve aplet can be used in conjunction with any of the functions available through the **MATH** menu, and can also reference any equations or functions defined in other aplets.

Example 3



"Find the value of a so that $\int_2^a x^3 - x\,dx = 4$"

Set **E1** to: $\int(2, A, X\wedge 3 - X, X) = 4$

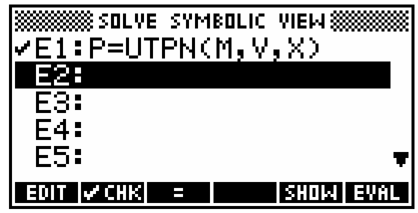
In the **NUM** view, set **A** to an initial guess of 3, and position the highlight on **A**. Ignore **X** since it is not really involved except as a temporary variable during the integration. Press **SOLVE** to obtain an answer (eventually) of 2.4495. The delay is caused by the repeated integrations as the calculator searches for better solutions. It is important to remember that the calculator does not use algebra in Solve – it uses an algorithm which is essentially a more sophisticated version of "guess, check & improve".

Example 4

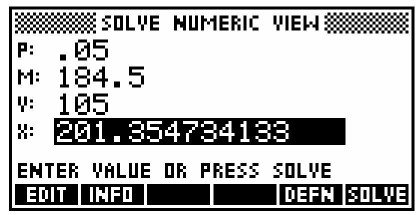
"Let X be a random variable, representing the heights of basketball players. If X is normally distributed, with $\mu = 184 \cdot 5$ and $\sigma^2 = 105$ then find the height which cuts off the tallest 5% of players."

The **MATH** function which allows you to work with the normal distribution is **UTPN** (see page 207) which gives the upper-tailed probability. The syntax is **UTPN(mean, variance, value)**.

In the Solve aplet, set **E1** to **P=UTPN(M,V,X)**.

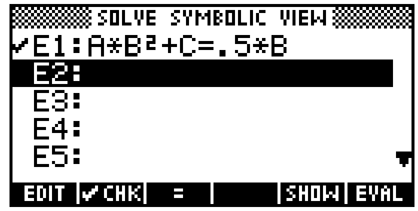


Enter the **NUM** view press SOLVE to obtain 201.35cm



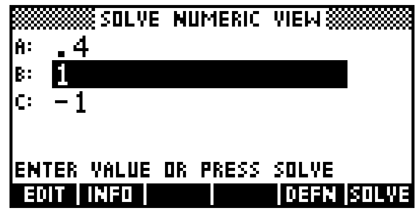
There are many more ways in which the Solve aplet can be used in conjunction with the functions available in the **MATH** menu. The **MATH** menu is discussed in detail starting at page 165, with worked examples which often use the Solve aplet.

A detailed explanation of PLOT in Solve

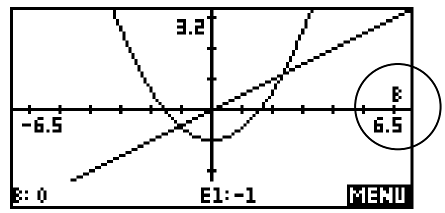


The **PLOT** view in the Solve aplet is a little more complex than most others, since the active variable (x, t, theta etc) changes according to the value for which you are trying to solve.

As an example, we will enter the equation $A*B^2+C=\cdot 5*B$ into **E1**. Suppose that we know the values of **A** and **C** but need to find **B**.

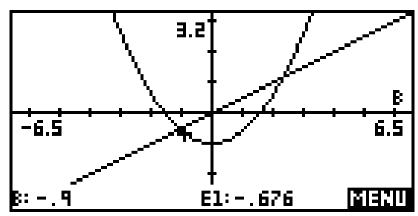


Now change to the **NUM** view and enter the values shown right. Ensure that the highlight is on **B** as shown and then press **PLOT**.

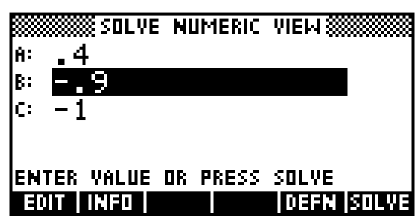


The result is a quadratic intersecting a line and the reason for this lies in how Solve interprets your equation.

When you select **B** by highlighting it, the calculator substitutes the supplied values in the **NUM** view into all other variables except **B** and graphs the left and right sides of the equation as two separate graphs. This may not always be obvious because the substitution may produce graphs which aren't visible on the default scale.



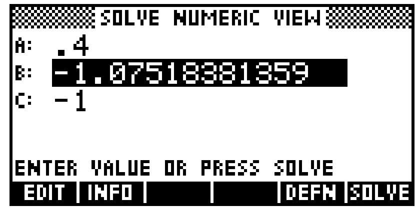
In the graph above you can see from the label on the horizontal axis (circled) that the active variable is **B**. In this case substitution means that $A*B^2+C=\cdot 5*B$ becomes $0\cdot 4B^2-1=0\cdot 5*B$. The left side is quadratic, while the right is linear.



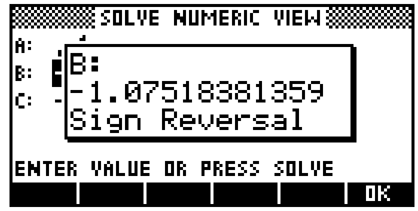
In this case there are clearly two points where the graphs intersect and hence where the left and right sides are equal.

Move the cursor near to the left hand intersection and then change back to the **NUM** view. When you do so, the approximate value you chose with the cursor is transferred as your first 'guess'.

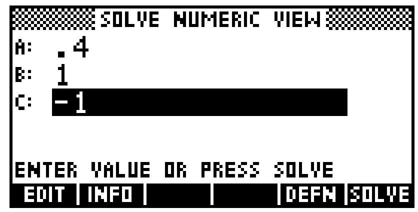
Now press **SOLVE** and you will see the calculator find the nearest solution to your guess. Finish by pressing **INFO** to verify that the solution is valid. See page 106 for more information regarding this.



Obviously the next step is to change back to the **PLOT** view, move the cursor near to the second intersection and **SOLVE** for that one too.

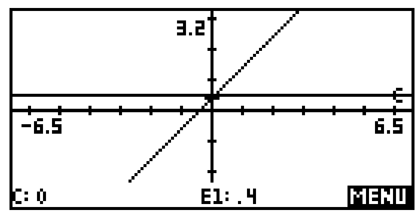


If we now change the active variable then there is an immediate change in the **PLOT** view to reflect this.

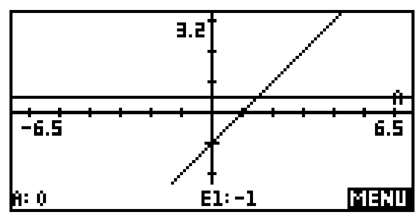


Change back to the **NUM** view and restore the variables to the original values, this time leaving the highlight on **C**.

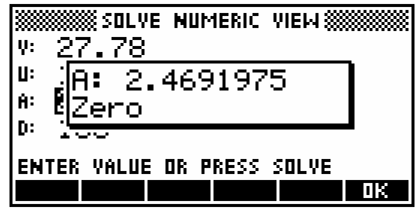
Since the active variable is now **C**, substitution of all other values means that $A*B^2+C=\cdot 5*B$ becomes $0\cdot 4+C=0\cdot 5$. In this case both sides are linear and this is shown in the **PLOT** view. Notice that the axis label is now **C**. This would have only one solution and there would be no need to supply any particular starting value.



If we duplicate this process for **A** then substitution of values means that $A*B^2+C=\cdot 5*B$ becomes $A-1=0\cdot 5$. Both sides are still linear, although not the same linear equations as before, and this is shown in the **PLOT** view. Notice that the axis label is now **A**.



The meaning of* INFO *messages



On pages 106, the values used were **V=** $27\cdot 78$, **U=** $16\cdot 67$ and **D=** 100 and we were solving for **A**.

Thus: $v^2 = u^2 + 2ad$
became: $(27\cdot 78)^2 - (16\cdot 67)^2 - 2\times a\times 100 = 0$ when substituted and re-arranged.

When you press INFO there are a number of possible positive responses. They are:

- ***Zero*** - The calculator tried to find a value of **A** which made this zero and, in the INFO message shown above, it is reporting that it succeeded.

- ***Sign reversal*** - This also indicates a correct solution, since normally one expects to find an answer of zero at a point where the re-arranged equation changes from positive to negative (or vice versa). 'Sign reversal' is a report that it couldn't get an answer to 12 significant digits that was precisely zero, just two answers minutely on either side of zero. This might indicate a discontinuity at the point but is more likely to indicate a satisfactory answer.

The only time that you need to worry is when you receive any of the messages below. These are:

- ***Extremum*** - it found a minimum, but could not reach zero. Try solving the equation $(x-2)^2 + 4 = 0$ and you will see this. The smallest value that $(x-2)^2 + 4 = 0$ can have is 4 at x=2, so the answer supplied will be very close to this (such as 2.000000001 or 1.999999999). The problem is that unless you check INFO you may not realize that this is not actually a valid solution.

- ***Bad Guess*** - the initial estimate you supplied was outside the domain of the function. For example, the equation uses a square root or a logarithm and you began from a value involving a negative.

- ***Constant?*** - no solution was found. The value of the function was the same at every point tested and wasn't the value you wanted.

Calculator Tip
It is critical that students recognize the ***Extremum*** case since it occurs quite often when the two sides of the equation approach closely but do not quite intersect. The student must at least recognize that the answer is invalid and, preferably, why.

This is the major reason why it is generally better to work in the Function aplet, since it allows you to see clearly whether or not there is a solution.

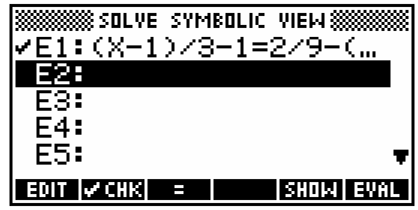
17

THE EXPERT: EXAMPLES FOR SOLVE

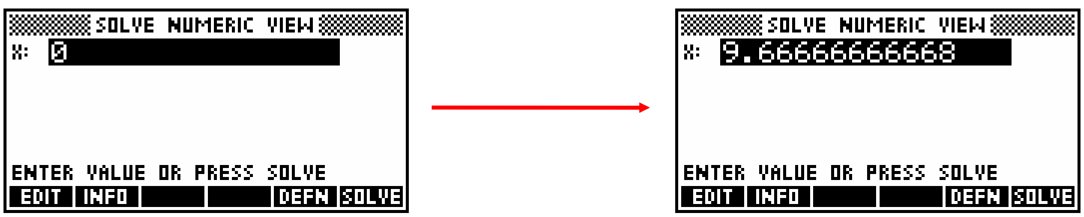
Easy problems

Have you ever thought "There has to be an easier way!" when confronted in a test with something like:

$$\frac{(x-1)}{3}-1=\frac{2}{9}-\frac{(3-x)}{4}$$



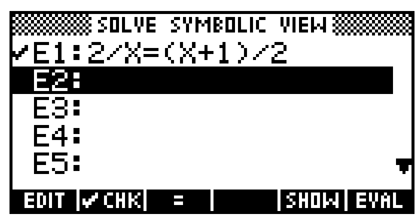
If you're sure there is only one answer to a problem, as there is in this case, then solving it is simply a matter of entering the equation into the **SYMB** view and solving it.

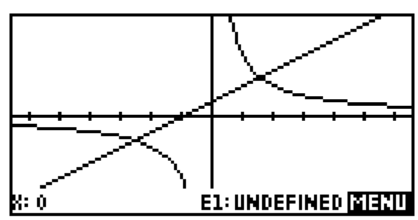


Harder problems

When you know or suspect that there is going to be more than one solution to a problem then the **PLOT** view can help you to obtain estimates.

For example: $\frac{2}{x}=\frac{(x+1)}{2}$





When you plot this you can see that the solutions are near -2 and 1·5. If we now switch to the **NUM** view we can use these as initial guesses. This will give solutions of -2·56 and 1·56.

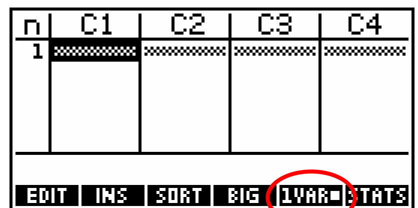
In this case the problem shown will graph easily on the default **PLOT** view but this will not always be the case. You may need to adjust the axes in **PLOT SETUP**. Your teacher may insist on an answer given as a surd, or on showing all working, but even in that case you'll at least be able to check your answer.

THE STATISTICS APLET - UNIVARIATE DATA

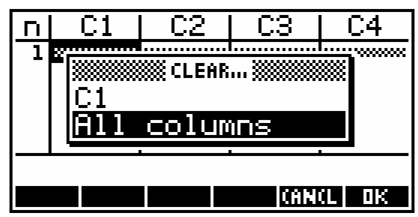
One of the major strengths of the hp 39gs & hp40gs is the tools they provide for dealing with statistical data. The Statistics aplet and its companion the Inference aplet provide very powerful yet easy to use tools with which to analyze statistical data.

The calculator treats univariate and bivariate data quite differently and those differences are reflected in the **SYMB** and **PLOT** views. Because of this the Statistics chapter has been split into two and univariate data will be dealt with first.

When you press START the Statistics aplet initially opens in the **NUM** view, offering easy input and editing of value. The **SYMB** view is reserved for specifying which columns contain data and which columns frequencies or, in the case of bivariate data, for indicating pairing of columns. If you have not already done so, go to the **APLET** view, highlight, RESET and then START the Statistics aplet.



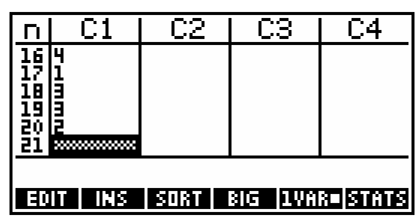
On the screen you will see a key labeled either 1VAR or 2VAR. Pressing the key under this label changes from univariate (1VAR) to bivariate (2VAR) and back. Make sure the key is showing 1VAR before proceeding.



If your **NUM** view had some data in it, you could press **CLEAR** (above **DEL**) and choose *All columns*. The **DEL** key is used to delete individual data points, rather than whole columns.

Let's use the following set of data and obtain all the usual statistics on it, and also plot a histogram and a box & whisker graph.

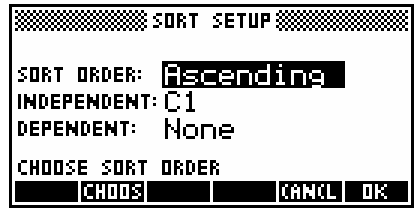
{ 2, 3, 1, 0, -2, 3, 4, 2, 2, 0, 6, 2, 3, 1, 0, 4, 1, 3, 3, 2 }



Move the highlight into column **C1** and enter the data, pressing the **ENTER** key after each piece of data.

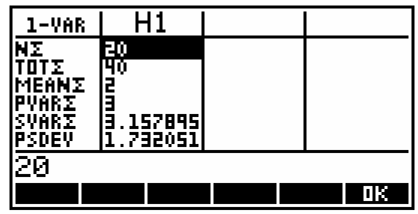
Looking at the bottom of the calculator screen you will see a series of tools provided for you. EDIT is not really worth bothering with because it is generally easier just to retype a number than it is to press EDIT and then use the arrow keys and **DEL** to change it.

The key labeled **INS** inserts space for a new number by shifting all the numbers down one space. **SORT** does exactly what it says… it sorts the data into ascending or descending order. The extra fields in the screen shot right are used with bivariate sorts or frequency tables and will be explained in that section of the notes. Press **CANCL** to stop the sort.

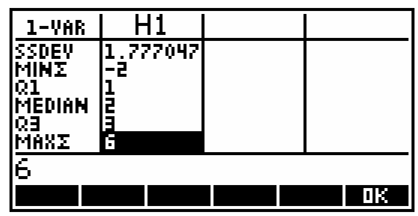


The **BIG** key provides access to a larger font size and **1VAR** vs. **2VAR** we have already discussed. The last key labeled **STATS** is the really useful one.

Making sure the highlight is in column **C1**, press the **STATS** key and you will see the screen shown right. If you use the down arrow, you can scroll down and see the rest of the screen (below right).



NOTE: If you get an error message instead of summary statistics, you may have forgotten to **RESET** the aplet before beginning this process. If the **SYMB** view defines columns which don't actually have any data in them then errors will result.



As you can see in the screens above right, the calculator gives not only the standard statistics that any scientific calculator would give, but also the minimum and maximum values, the median and the upper and lower quartile cutoffs. The mode is not given, but this is easily obtained from the histogram as we will see later.

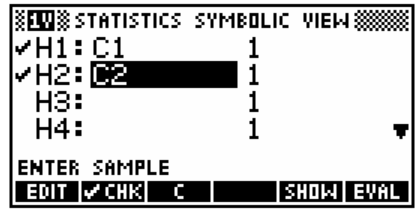
Let's create a second column of data with all its values exactly double the values in the first column. We can use the **HOME** view to avoid having to retype the values as follows…

- Change to the **HOME** view and type the command shown right, then press **ENTER**. The **STO▶** command is found on the screen keys at the bottom of the screen.

- Press the **NUM** key to change back to the **NUM** view. You should find your new column created and ready.

If you now use the **STATS** key, you will find that you still only see statistics for the first column (**H1**). The reason for this is that you have not set up the second column in the **SYMB** view. The default setup is to recognize column 1 as 'in use' and so we didn't need to do this before now.

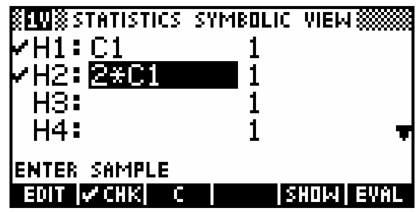
Change into the **SYMB** view and edit yours so that it looks like the one on the right. You must make sure that **H2** is ✔CHKed, because only checked columns will show in the STATS view.



Note that a screen key is provided to give you the letter **C** without having to use the **ALPHA** key.

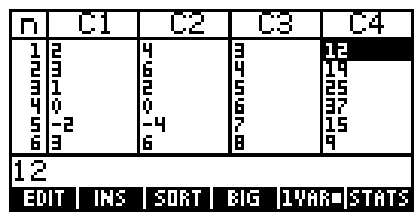
The stats are should now be available for both columns of data. Return to the **NUM** view and press STATS to see them.

As an alternative to the method used above, you could also have created a 'virtual' column of data by using the **SYMB** view. Simply enter an expression giving the new column as a function of the old one. Although the 'virtual' column will not be displayed in the **NUM** view (and does not consume memory), it can now be graphed and analyzed statistically.



You may be wondering why the **SYMB** view is organized around histograms **H1, H2..H0** rather than simply around the columns **C1, C2..C0**. The reason is that it allows you to easily cope with a frequency table by setting up one column to represent values and another to represent the frequencies. If you are not using a frequency table then the frequencies are normally set, by default, to 1 as can be seen on the previous page.

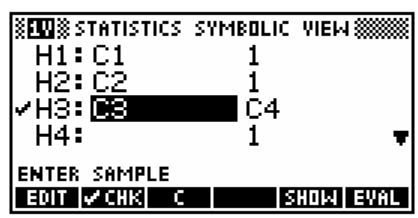
So how do we deal with frequency tables? Let's set up columns **C3** and **C4** to represent the table below.



x_i	freq.
3	12
4	19
5	25
6	37
7	15
8	9

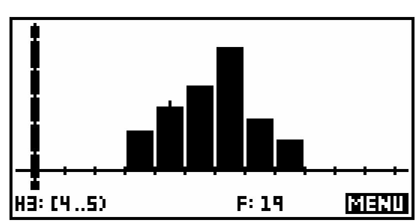
Enter the values 3, 4, 5, 6, 7 and 8 into **C3** and then 12, 19, 25, 37, 15 and 9 into **C4**.

Now set up your **SYMB** view to look like the one right. Make sure only **H3** is checked. You can, if you wish, delete the definitions for **H1** and **H2** but make sure they are not ✔CHKed.

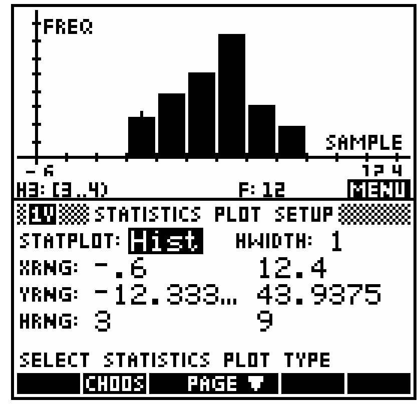


The reason for the last instruction is that only one histogram can be drawn at a time and if more than one data set is ✔CHKed then only the first one is drawn.

Now use **VIEWS** *Auto Scale* to plot the graph. You will hopefully find that it looks like the one on the right. The *Auto Scale* function is always very effective in the Statistics aplet and is recommended.

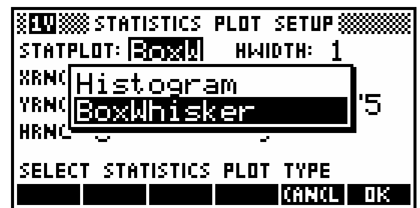


If you use the left/right arrows and look at the bottom of the screen you'll see that the frequencies and ranges are listed. It is probably worth tidying up this graph up a little by going into **PLOT SETUP** and (on the second page) setting the **YTick** value to be 5 instead of 1. In the graph to the right the **Labels** option has also been ✓CHKed.

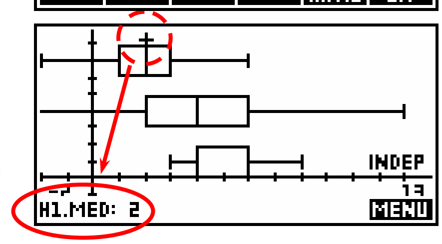


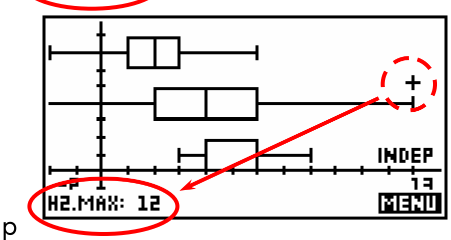
You probably noticed a lot of other options in the first page of **PLOT SETUP**. Their explanations follow.

The setting of **Statplot** controls what type of graph is drawn. There two choices are *Hist* (short for histogram) or *BoxW* (Box and Whisker). Pressing the [+] key while **Statplot** is highlighted will switch between these two, or you use the CHOOS button to pick from a menu.



Unlike histograms, it is possible to have more than one box and whisker graph plotted. This makes comparisons between data sets very easy. If you look for the cursor (circled) in the diagram shown right, you will see that when TRACE is turned on then information about the graph is given at the bottom of the screen. As usual the up/down arrows change from graph to graph, while the left/right arrows move within the graph.

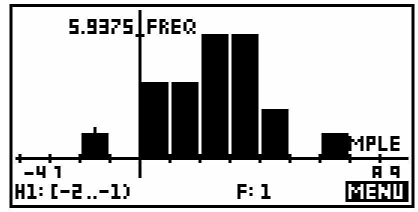




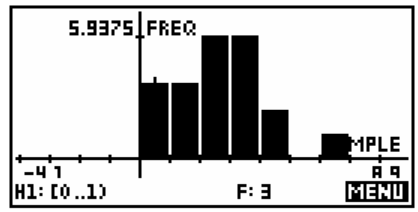
As an aside, pressing the MENU key produces the normal tools of ZOOM, TRACE and DEFN. They all behave in the normal manner as was discussed in detail in the Function aplet chapter. The DEFN tool can be quite useful by displaying information on which columns make up each graph if you lose track.

Looking again at the screen shot of the first page of **PLOT SETUP** (near the top of the page) you will see that there are three ranges. As with the Parametric and Polar aplets, **XRng** and **YRng** control how much of the graph is seen. If your histogram has frequencies of (say) up to 30 then you need to make sure your **YRng** reaches at least that value or the top of the histogram will be cut off. In the same way, if your data has a minimum value of -5 and a maximum of 35 then your **XRng** will need to cover at least that range of values. This is normally not worth worrying about since using **VIEWS** *Auto Scale* generally produces very satisfactory results.

The effect of **HRng** is rather different. It controls what range of data is displayed on the graph, regardless of what axes are used. It is normally set automatically to be the maximum and minimum values for the data. For example histogram **H1** (shown right) has an **HRng** of -2 to 7. If **HRng** were changed in **PLOT SETUP** to 0 to 7 then the graph will lose the left column representing the value of -2 as shown below.



The advantage of this is that it allows you to eliminate outliers from your graph quite easily. However, eliminating them from your graph does not eliminate them from inclusion in the calculation of the values that appear in the STATS page. To do that you would need to delete the actual data itself.



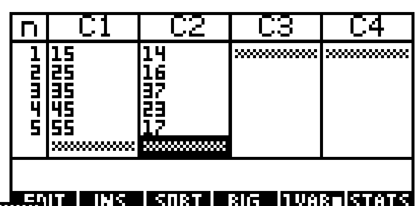
data	freq
10 - 19	14
20 - 29	26
30 - 39	37
40 - 49	23
50 - 59	17

One final note concerns grouped data. We saw earlier how to deal with data displayed in a frequency table, but did not deal with the case where the data was also grouped into intervals or classes.

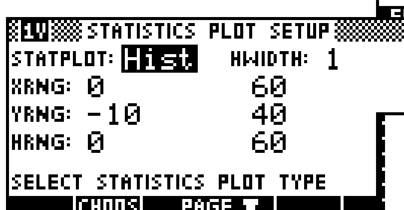
For example, suppose we want to analyze the set of grouped data in the table on the left.

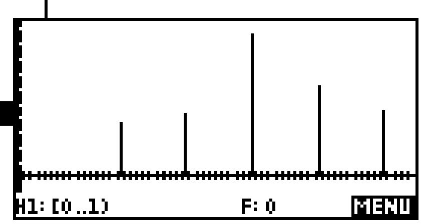
As with most calculators, the hp 39gs & hp 40gs provide only limited methods to deal with data of this form. Summary statistics can be obtained by entering the mid-points of the intervals as the data values but these will only be approximations, as nature of the data itself does not allow calculation of exact values.

The problem with using the mid points is that attempting a normal **PLOT** will produce a series of isolated columns, each with width one unit. The reason for this is that the calculator is assuming that there are 14 values of exactly 15 (the mid-point) because it doesn't realize that the columns extend the width of the interval 10 - 19.

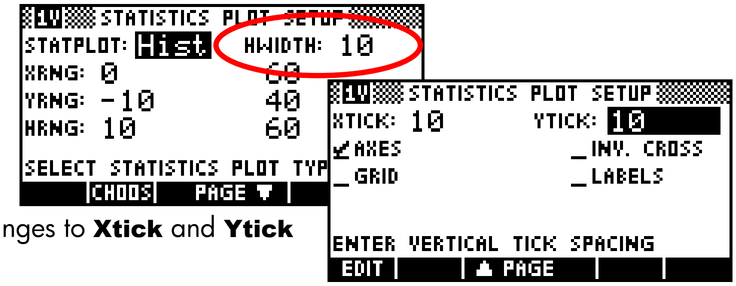


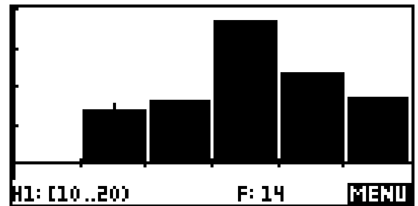
However this can be fixed by using the setting **HWidth** as outlined on the next page.



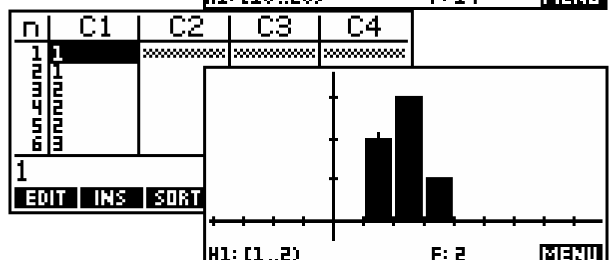


The **HWidth** variable controls the width of the columns, with the initial starting value and end value set by **HRng**. In the frequency table on the previous page the interval width was 10. By setting **HWidth** to 10 in the **PLOT SETUP** view as shown right we can produce the graph shown below. Note the changes to **Xtick** and **Ytick** also.

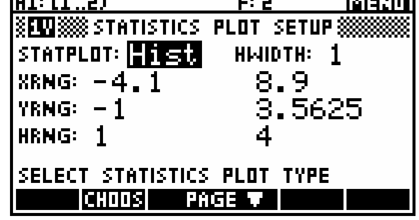




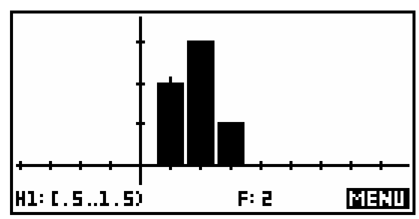
Another use for the variables **HWidth** and **HRng** is to re-orient the columns so that they are centered on the data point. For example, the data set below produces the graph to the right of it using *Auto Scale*.



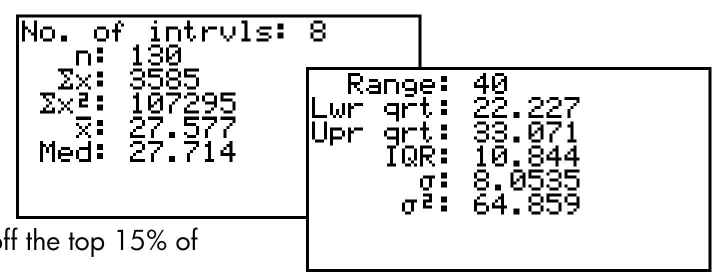
You may have noticed that the columns are not centered around the data points 1, 2 and 3. Instead the columns extend from 1..2, 2..3 and 3..4. This bothers some people and can easily be fixed by changing the **HRng**. If you look at the **PLOT SETUP** values you will see that they are currently set to 1 to 4. This is the default behavior for *Auto Scale*.



To center the columns you need only change the minimum value for **HRng** by subtracting a half column width giving 0.5. The results are shown right and you can see that the columns are now centered as required.



If you want to use linear interpolation to obtain better values for the median and quartiles then you can download an aplet from the author's website called "Grouped Data". This will calculate the values of the medians and quartiles using linear interpolation and will also allow calculations such as "What value will cut off the top 15% of data?".

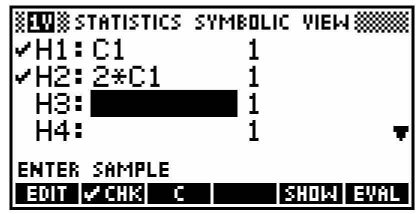


THE EXPERT: SIMULATIONS & RANDOM NUMBERS

New columns as functions of old

You have already seen the use of one trick when we created a new column **C1** by storing **2*C1** into **C2** using the **HOME** view. This can be used to create new columns as functions of any number of others. For example, a set of data that you suspect is exponential could be 'straightened' by storing **LN(column)** into a fresh column. Changes of scale and origin can be investigated in this way by storing (say) **-2*C1 + 3** into **C2**. You can even combine columns such as **C1 + C2▶C3**.

If you don't particularly need to *see* the data as a fresh column, you can use the **SYMB** view to accomplish the same thing in a simpler way. For example, the **SYMB** view snapshot on the right would accomplish the same thing as storing **2*C1** into **C2**. A histogram of **H2** would look the same as the one we produced earlier using the **HOME** view, and the **STATS** command would give exactly the same results. The advantage of this is that it takes much less memory if both columns need not be stored.



For the teacher this can be a handy way to create sets of data that conform to particular models. For example, if you would like a set of data that conforms to a model of $\hat{y} = 2.5e^{0.03x}$ then simply enter any values randomly into **C1** and then, in **HOME**, perform the calculation **2.5*e^(0.03C1)▶C2**. Of course, the result will be a set of y values which exactly match the model and this is not desirable. The teacher should now go through them and introduce some random error so that they are no longer a perfect fit.

Simulating Dice

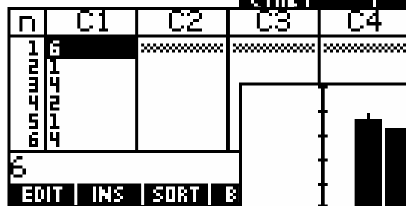
The most common experiments in probability involve the rolling of dice. This can be simulated in the Statistics aplet using the **MATH** menu function **MAKELIST**. (For more detail see page 190)

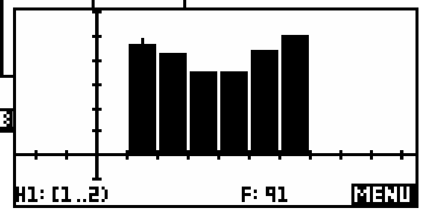
The syntax is: **MAKELIST(expression, variable name, start, end, increment)**

where *expression* is the mathematical rule used to generate the numbers.
variable name is the letter (X, Y etc.) that is to be used in the expression (any other letters will be taken as constants).
start is the first value *variable name* is to take.
end is the upper bound for *variable name*.
and *increment* is the amount that variable name should be incremented by in each iteration.

For example: **MAKELIST(X2,X,1,10,2)**
would produce { 1, 9, 25, 49, 81 } as **X** went from 1 to 3 to 5 to …

Similarly the expression **INT(RANDOM*6+1)** will simulate one roll of the die. This means that **MAKELIST(INT(RANDOM*6+1),X,1,500,1)** will simulate 500 rolls of a normal die. We therefore need only store the resulting list into a Statistics aplet column to analyze and graph it. This is shown in the series of screen shots to the right.





To simulate the adding of two six sided dice, use the similar expression **MAKELIST(INT(RANDOM*6+1)+INT(RANDOM*6+1),X,1,500,1)**

Simulating Random Variables

It is occasionally handy to be able to simulate a set of observations on random variables of various sorts. This can also be done using the **MATH** menu function **MAKELIST**. See the previous page for the syntax of **MAKELIST**. It is also covered in more detail on page 190)

Example 1: Simulate 100 observations on a Uniform[0,1] random variable to be stored in **C2**.

In the **HOME** view type: **MAKELIST(RANDOM,X,1,100,1)▶C2**

Note: 1. The ▶ symbol is one of the screen keys in the **HOME** view, appearing as **STO▶** and read as 'store'.
2. The **X** is simply a dummy variable here to count off the values.

Example 2: Simulate 50 observations on a discrete uniform[3,7] random variable

In **HOME** type: **MAKELIST(INT(5*RANDOM+3),X,1,50,1)▶C2**

Example 3: Simulate 50 observations on a Binomial random variable with n = 20 and p = 0.75.

In **HOME** type: **MAKELIST(Σ(I=1,20,RANDOM<0.75),X,1,50,1)▶C2**

Note: The Σ and the = are both on the keyboard. This will be a relatively slow calculation because it involves evaluating 1000 random numbers.

Example 4: Simulate 100 observations on a normal random variable $N(\mu=80, \sigma^2=50)$.

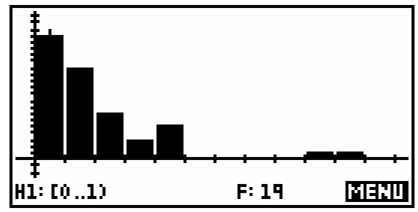
Ensure that **MODES** is set to radian measure and type:

MAKELIST(80+√50*(√(-2*LN(RANDOM))*sin(2*RANDOM)),X,1,100,1)▶C2

Example 5: Simulate 50 observations on an exponential distribution with mean = 2.

In the **HOME** view type: **MAKELIST(-2*LN(1-RANDOM),1,50,1)▶C2**

As an illustration, the result of this particular simulation is shown graphically on the right. Its mean turned out to be 2.067 (3 decimal places.). Yours will be different of course - after all, that's the point of using *random* numbers!



Calculator Tip

The **RANDOM** function is not truly 'random' any more than it is on any computer. If you use the **RANDOM** command repeatedly on two calculators just out of the box then you will see the same set of numbers on both calculators.

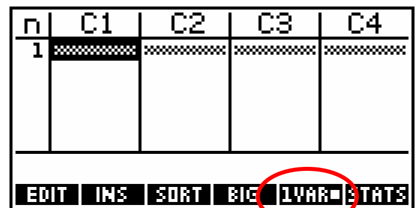
This is because the **RANDOM** function uses a 'seed' value and this is the same initially for all calculators. It is possible to fix this by 'seeding' a new number using the **RANDSEED** command. For information on this see page 206.

THE STATISTICS APLET - BIVARIATE DATA

As mentioned in the Univariate section, one of the major strengths of the hp 39gs & hp 40gs is the tools they provide for dealing with statistical data. Unlike the others, the Statistics aplet begins in the **NUM** view which offers easy input and editing of values, while the **SYMB** view is reserved for specifying which columns contain data and which ones frequencies, as well as for indicating pairing of columns for bivariate data.

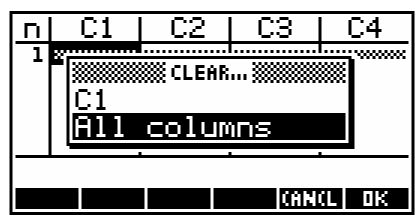
The hp 39gs & hp 40gs treat univariate and bivariate data quite differently and those differences are reflected in the **SYMB** and **PLOT** views. Because of this the Statistics chapter has been split into two.

The previous section dealt with univariate data and we are now going to look at bivariate (paired) data. On the screen in the **NUM** view of the Statistics aplet you will see a key labeled as either **2VAR** or **1VAR**. Pressing the key under this label changes from univariate (**1VAR**) to bivariate (**2VAR**) and back. Make sure the key is showing **2VAR** before proceeding.



x_i	y_i
1	5
3	10
8	18
5	13
7	16
3	8
6	15
2	8
7	18
9	22
8	17
5	15
7	14
6	18
8	20
5	12
2	8
0	4
7	17
8	19

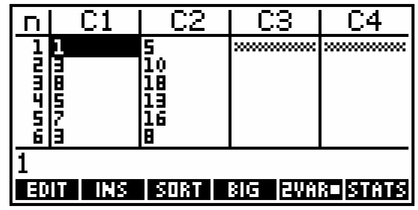
If your **NUM** view already has some data in it, press **SHIFT CLEAR** (above **DEL**) and choose *All columns*.



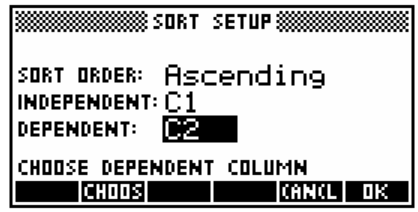
The **DEL** key is used to delete individual pieces of data, rather than whole columns. One must be careful when doing this for bivariate data. If your data pairs were in column 1 (**C1**) and column 2 (**C2**) then deleting a single piece of data from **C1** without deleting its paired element in **C2** would destroy the relationship between the two columns.

In this chapter we will use the set of data left to show the use of the bivariate features of Statistics. We will obtain all the usual statistics on it, and draw a scatter graph.

Move the highlight into column **C1** and enter the x_i values, pressing the **ENTER** key after each one. Then do the same for the y_i values in **C2**. The result is shown right.



Looking at the bottom of the screen you will see a series of tools provided for you. As before, EDIT is not worth bothering with. The key labeled INS inserts space for a new number by shifting all the numbers down one space. The BIG key provides access to a larger font size (for us old fogeys) and 1VAR vs. 2VAR we have already discussed. The last key labeled STATS is the really useful one.



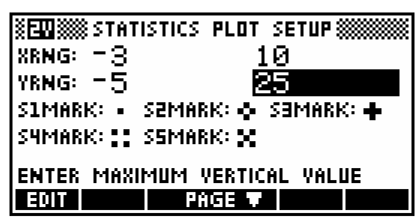
The SORT option is capable of dealing with bivariate data if you are careful to enter the column number of the dependent column into the appropriate space. The CHOOS key will pop up a list of columns from which to choose, or you can use the **ALPHA** key to type in the column name.

Calculator Tip

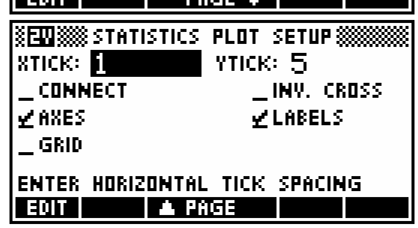
You can enter the x_i and y_i data into both columns simultaneously if you enter it as ordered pairs in brackets.

i.e. as **(1 , 5)** **ENTER** **(3 , 10)** **ENTER** etc.

Returning to the data from the previous page, having entered it into the **NUM** view we could now use **VIEWS** and *Auto Scale* to produce a plot (this generally produces very satisfactory results), but let's have a look at the **PLOT SETUP** screen instead.

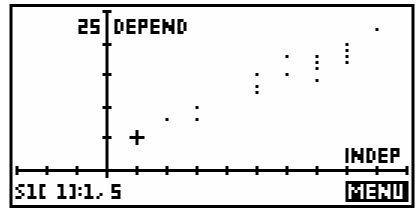


As you can see, it is very similar to the other **PLOT SETUP** screens that we have encountered with the main difference consisting of the list of settings for **S1MARK**, **S2MARK** etc. These are the markings that are to be used in plotting the points, allowing you to choose different markings for different data sets if you are graphing multiple data sets.

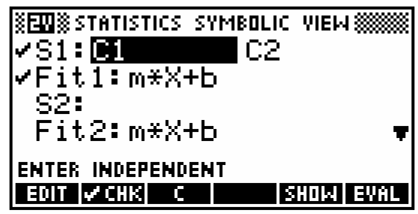


Set your **PLOT SETUP** screen so that it looks like the one shown above right, also switching to the second page and ensuring that it is checked as shown right and setting **YTick** to 5. Personally I usually change the **S1MARK** to a larger symbol than the simple dot because I find it easier to see than single dots.

If you now press **PLOT** you will see the result shown right. If you look at the screen you will see a small cross and, at the bottom of the screen, a listing of **S1[1]: 1,5**. This is telling you that the cross is currently on the first point in data set **S1** whose value is **(1,5)**. Using the left/right arrow keys you can move this cross through the data set with the values being reported at the bottom of the screen.



If you have more than one data set displayed on the screen then the up/down arrows move from one set to the other, unless the fit line is also showing in which case the cursor moves first to the fit line and then to the next set of data. Information is given later on how to display the fit curve and to choose the type of curve.

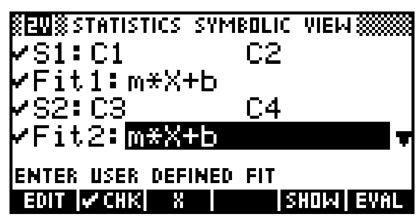


Although we have not mentioned it in the example so far, part of the procedure in dealing with bivariate data is to specify the relationships between columns before plotting. This is done via the **SYMB** view as it is for univariate data. The default setting is for the data to be in columns **C1** and **C2** and because that's where our data is we were able to bypass this process.

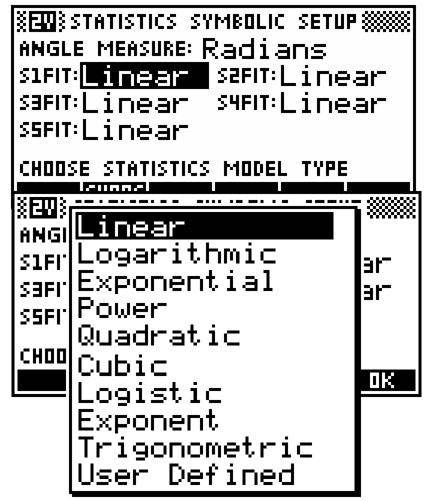
The **S1:**, **S2:** … refer to data sets 1, 2… This allows you to display more than one set of bivariate data by specifying the columns for each set. Columns can be used in more than one set. Note that a **C** screen key is provided for your convenience.

The screen above shows the default setup when you **RESET** the aplet in the **APLET** view. It specifies that columns **C1** and **C2** are paired and that a linear fit (**m*X + b**) is to be used when calculating a line of best fit.

If you intend to use other columns then you must register them in the **SYMB** view. For example, to use data entered into **C3** and **C4** you must register them as shown right.



Choosing from available fit models



The Statistics aplet is the only one which has a **SYMB SETUP** view, and even then only in **2VAR** mode. This view is supplied to allow you to specify what type of fit equation is to be used. The choices are:

Linear - **m*X+b** $(Y = mX + b)$

Logarithmic - **m*LN(X)+b** $(Y = m\ln(X) + b)$

Exponential - **b*EXP(m*X)** $(Y = b\,e^{mX})$

Power - **b*X^m** $(Y = b\,X^{m})$

Quadratic - **a*X^2+b*X+c** $(Y = aX^2 + bX + c)$

Cubic - **a*X^3+b*X^2+cX+d** $(Y = aX^3 + bX^2 + cX + d)$

Logistic - **L/(1+a*exp(-b*X))** $(Y = \dfrac{L}{1 + a\,e^{-bX}})$

This fits the data to a logistic curve where L is the saturation value. See tip below.

Exponent - **b*m^X** $(Y = b \times m^{X})$ This model is essentially the same as the *Exponential* version but without the use of *e*. This caters for students who have been exposed to exponential equations but not to the extent of *e*.

Trigonometric - **a*SIN(b*X+c)+d** $(Y = a\sin(bx + c) + d)$

This model fits a possible trigonometric curve to the data. Because the sine curve is periodic the answer will not be unique.

User Defined - discussed on the following page.

Calculator Tip

1. If you want the value of L calculated automatically for the Logistic model then store a value of zero into **L** in **HOME**. If the value is known, you can store a positive real value into memory L prior to the curve fit and this will be used.
2. If you calculate a line of best fit and want to remove the resulting equation from the **SYMB** display and return to the **m*X+b** display then just position the highlight on the relevant **Fit:** line and press the **DEL** key. When you do this, the **✓CHK** will be removed and will have to be reset.

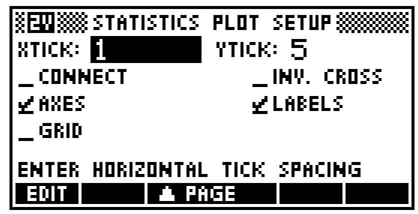
The User Defined model

When you set the model to user defined it means that you are expected to supply the complete equation, including the values of any coefficients. The calculator will not calculate the values of any variables you include. For example, if you were to supply an equation of **A/(X–B)** as your model then the calculator would use the values of **A** and **B** currently in memory. By repeatedly adjusting the values of **A** and **B** in **HOME** you could find the best version of the curve for the data.

This may seem to be a useless model but it can be quite useful if you want to experiment with a model that is not one of those supplied with the calculator.

Connected data

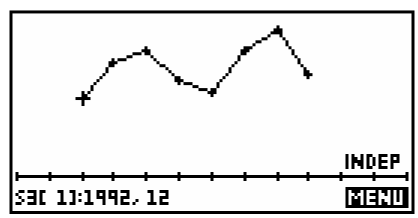
One of the settings on the second page of the **PLOT SETUP** screen can be useful for some types of data. For example, one of the common tasks in many mathematical courses is the analysis of time series data. Unlike most bivariate data, time series values are usually plotted as a line graph - i.e. as connected points. This facilitated by **Connect**.



For example:

The sales of an ice-cream shop are shown as quarterly sales figures for the years 1992 to 1993. Display this data on a line graph.

Yr	1992				1993			
Qtr	1	2	3	4	1	2	3	4
Sales ($1000)	12	18	20	15	13	20	23	16

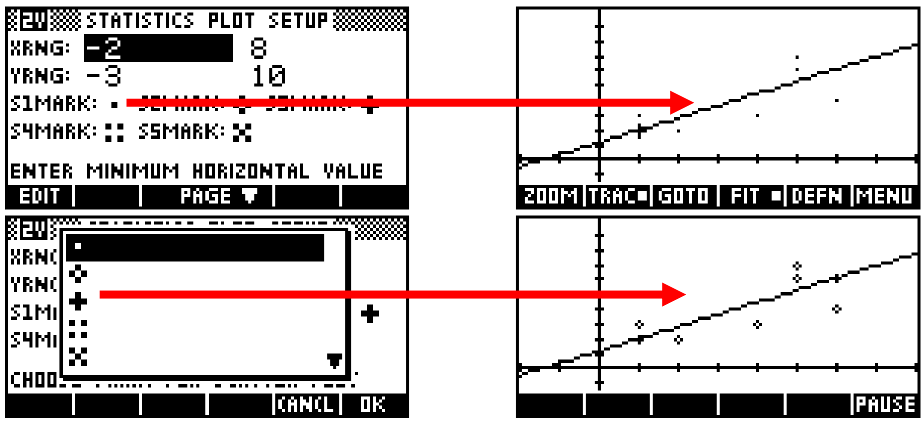


The result (with **Connect** checked) is shown right. Notice that although the **X** values in the table were 1, 2, 3, 4, 1, 2, 3, 4 we can't use them that way because it would result in 1992 Qtr 1 being graphed on top of 1993 Qtr 1. The graph on the right was produced by entering values of 1992, 1992.25, 1992.5, 1992.75 etc.

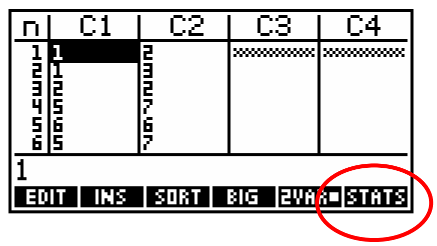
This is not the end of the story though. When you activate **Connect** the calculator will always sort the points into ascending order based on their x values before graphing. Thus (1,3) (5,4) (2,7) would be graphed as (1,3) (2,7) (5,4). Not doing this would result in a graph that doubled back on itself and looked like a scribble instead of a line graph.

Calculator Tip

If you have trouble seeing the small dots that the calculator uses in its scatter-graphs by default then you will be interested in the settings circled on the right. If you move the highlight onto the mark for the data set you are using and press **CHOOS** then you will see the menu shown below from which you can choose a different mark. The contrast is illustrated below.



Two Variable Statistics

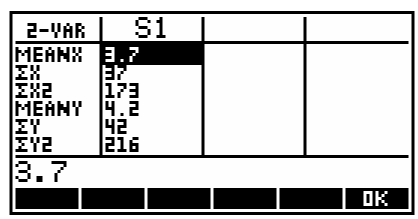


As with univariate statistics, summary statistics are available through the **STATS** key in the **NUM** view.

Pressing this key changes to a screen which lists the following:

MEANX, **ΣX**, **ΣX²**, **MEANY**, **ΣY**, **ΣY²**, **ΣXY**, **SCOV**, **PCOV**, **CORR** and **RELERR**.

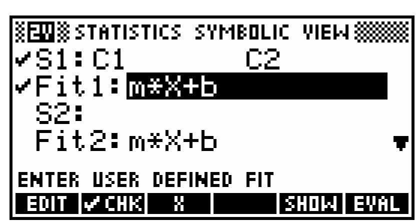
See page 131 for information on **RelErr**.



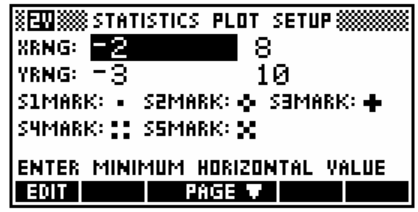
x_i	y_i
1	2
1	3
2	2
5	7
6	6
5	7
4	3
5	6
6	4
2	2

We will use the set of data shown in the table on the left to go through the analysis process a second time and this time examine the line of best fit.

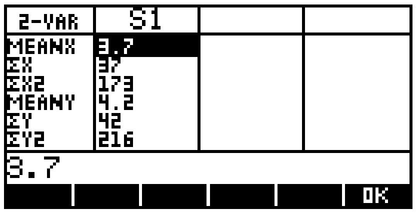
Enter the data into the **NUM** view and then switch to the **SYMB** view. Make sure that **S1** (data set 1) is set to **C1** and **C2**, **✓CHK**ed, and that the fit is linear. If not, the fit can be changed in the **SYMB SETUP** view.

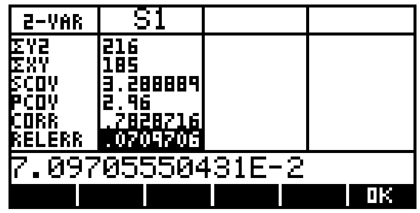


Now change to **PLOT SETUP** view and set the axes as shown right.



From the **NUM** view, press the **STATS** key and you will obtain the results listed in the screens shown below.

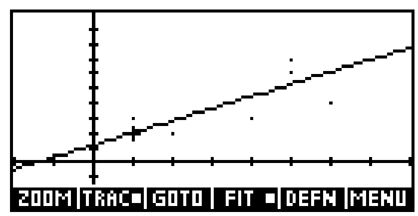




Calculator Tip

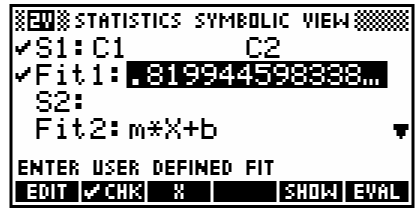
Make sure that your data set is defined and **✓CHK**ed in the **SYMB SETUP** view before you try to obtain these results. Results are only given for data sets that are defined and **✓CHK**ed.

If you now press the **PLOT** key you will see the graph shown right. If there is no line of best fit on yours, press the **MENU** key to get the list of programmable functions along the bottom of the screen and then press the **FIT** key.

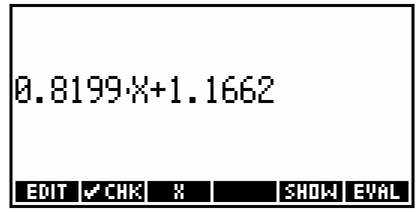


As soon as the line of best fit has been obtained, you can switch back to the **SYMB** view to get its equation. This equation is only calculated when called for by pressing the **FIT** key.

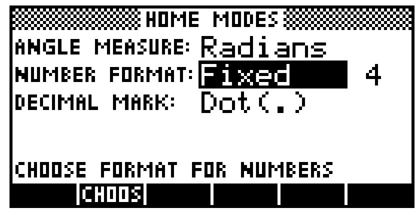
In the **SYMB** view (see right) the equation is given to so many decimal places that it doesn't fit onto the screen. The simplest way to see the entire equation is to position the highlight on the equation and press the **SHOW** key.



When you do you will obtain the view seen right which gives the equation of the line of best fit as $\hat{y} = 0.8199x + 1.1662$. With a correlation coefficient of 0.7829 (from the summary stats seen previously) this would probably not be regarded as reliable.



On the screen above the equation is given to 4 decimal place accuracy. This is accomplished by changing to the **MODES** screen (above the **HOME** key) and specifying **Fixed 4** as the **Number Format**. If you don't do this then the alternative is to use the arrow keys to scroll the equation left or right if it extends off the screen.



Calculator Tip

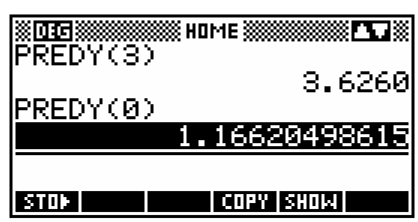
One of the most common mistakes people make is to press **EDIT** instead of **SHOW**. Always use **SHOW** when you want to display the entire equation. An explanation is given below.

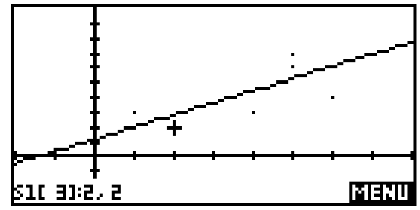
It is very important that you display the equation in the **SYMB** view by pressing **EDIT** instead of **SHOW**. Although pressing **EDIT** displays the equation in a form which can be scrolled through, the problem with this method is that the calculator thinks that by editing it you are choosing to over-ride its choice of equation with your own. As long as you press **CANCL** there is no problem but if you press **ENTER** it changes the **FitType** from whatever you had previously chosen to *User Defined* instead.

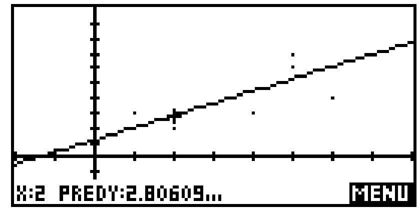
If you are not aware that this has happened then it can have very serious consequences. For example, if you were now to change the data and then re-plot, the fit equation will not be re-calculated and will become inaccurate. Any predictions made from it would be incorrect.

We can make predictions from our line of best fit in two places - the **HOME** view and the **PLOT** view.

In the **HOME** view we use the functions **PREDY** and **PREDX** from the Stat-Two section of the **MATH** menu. The functions **PREDX** and **PREDY** use whatever was the last line of best fit calculated. It is up to you to ensure that the one you want used was the one last graphed. If I want to predict a y value for x = 3, then I simply type **PREDY(3)** into the **HOME** view as shown right. Many people choose to simply type '**PREDY**' using the **ALPHA** button instead of going through the **MATH** menu.







Using the **PLOT** view is the probably the more visually appealing method of obtaining predicted values. When you have plotted a set of data and its fit curve then pressing up arrow will change the focus of the values at the bottom of the screen from the data points to the **PREDY** values. In the screen snapshots shown right the focus changes from data point **3:(2,2)**, to the **PREDY** value for x = 2 of 2.806.

If there is more than one data set (and fit lines) graphed then the up arrow will move progressively from one to another and finally back to the first.

Pressing left and right arrows will move along the fit line but only on pixel positions, which may not be suitable if the scale is not chosen carefully. A better way is to use the **GOTO** key to obtain **PREDY** values for any required value of *x*, including values which would normally be off-screen.

Another aspect of bivariate stats needs to be remembered when the fit chosen is not linear. Mathematically, the correlation coefficient is a strictly linear measure of the goodness of fit and this means that the correlation coefficient quoted in the **STATS** view is *always* for the linear model even when some other model is chosen.

There are two methods of dealing with this. The first is to use another measure of goodness of fit. The second is to 'linearize' the data (discussed on the next page). The calculator provides an alternative measure of goodness of fit via the **RelErr** value in the **STATS** view.

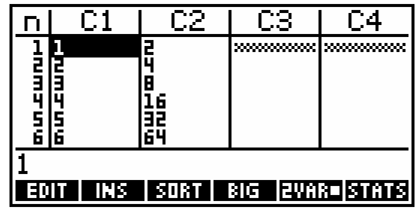
x_i	y_i
1	2
2	4
3	8
4	16
5	32
6	64

$$\text{RelErr} = \frac{\sum_{i=1}^{n}(y_i - \hat{y})^2}{\sum_{i=1}^{n} y_i^2}$$

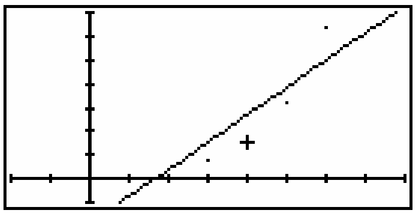
RelErr is defined as the measure of the relative error in predicted values when compared to data values, and has the formula shown right. The smaller the value of **RelErr** the better the fit. The $\hat{y}$ values are obtained using the **PREDY** function internally. The only drawback to **RelErr** is that there is no upper limit its value of as there is for the correlation coefficient. The interpretation placed on it is that the closer it is to zero the better the model fits the data. This value is available for any of the data models, including the user defined model.

Alternatively, when data is non-linear in nature you can transform the data mathematically so that it *is* linear. Let's illustrate this briefly with exponential data.

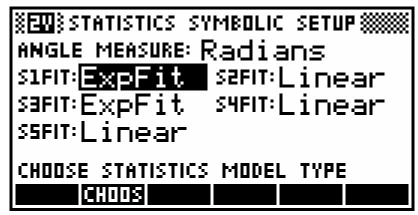
As you can see, I chose a very simple rule for the data of $y = 2^x$.



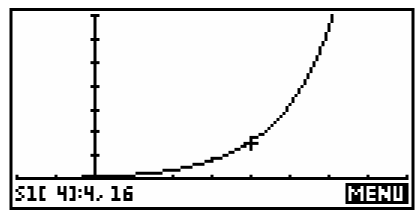
If you set up a linear fit for the data in **S1**, and then view the bivariate stats, you will find that the correlation for a linear fit is 0.9058



As you can easily see from the graph left, a linear fit is not a very good choice.

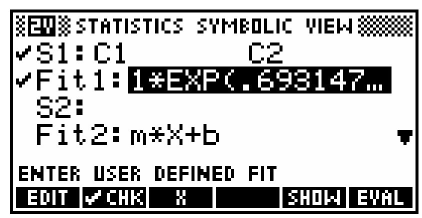


If we change now to the **SYMB SETUP** view and choose an Exponential fit rather than a linear fit then the results are far better.



The curve which results in the **PLOT** view is exactly what is required and the equation comes out as $Y = 1 \cdot EXP(0.693147X)$

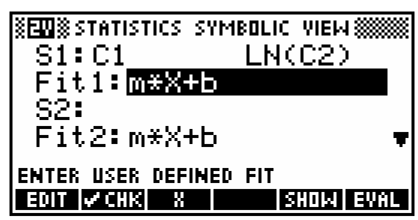
This "**EXP(**" is the calculator's notation for $Y = 1 \cdot e^{0.693147X}$ which then changes to $Y = 2^X$.



Checking the **STATS** key shows that the correlation is unchanged at 0.9058 even when the new equation clearly fits the data perfectly.

The value of **RelErr** on the other hand has changed from 0.09256 for the linear fit, to a value very close to zero for the exponential model (rounding error may result in something non-zero).

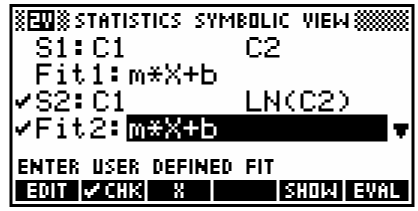
The alternative to using **RelErr** is to graph column **C1** against **ln(C2)** which also straightens the data.



'Linearizing' will cause problems if some of the data points are outside the domain of the function you use, such as negative values in a log function. On the other hand, you have far more control if you are able to choose the exact function. For example, if you had a set of data which was derived from cooling temperatures then you would probably find that it was asymptotic to room temperature rather than the x-axis. The built-in equation assumes that the data is asymptotic to the x axis and would not give a good fit. You could get better results by subtracting a constant from the whole column first.

THE EXPERT: MANIPULATING COLUMNS & EQNS

New columns as functions of old



As with univariate statistics, you can use functions of old columns as new sets of data. See the Univariate version of this section for two different ways of doing this.

For example, a set of data (**C1,C2**) that you suspect is exponential could be straightened by setting up **S2:** as (**C1,LN(C2)**).

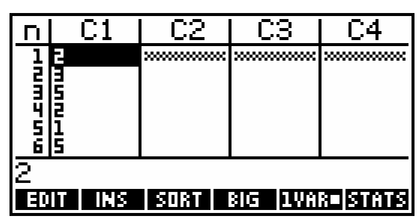
The effects of changes of scale and origin on data and summary statistics can be investigated in this way by storing, for example, **-2*C2+3** into **C2**. You can even combine columns in this way, such as storing **C1+C2** into **C3**.

Using values from STATS in calculations

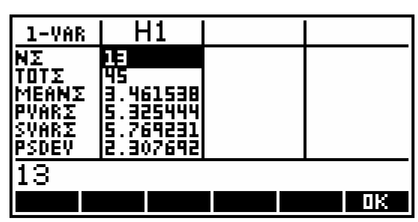
It is often useful to be able to retrieve values such as the mean and standard deviation for use in further calculations. With most simpler calculators these values are found by pressing keys rather than reading from a STATS screen, so doing a calculation like 'multiply the mean by 3.5' is not hard. The values shown on the STATS screen can also be retrieved for use on the calculator relatively easily.

For example, the set of data below contains a suspected outlier (erroneous value). In this case one might suspect a missing comma between the last two values.

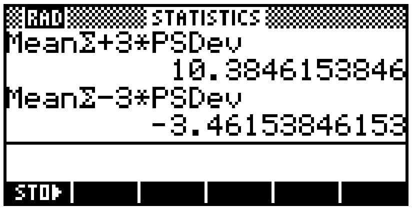
{2, 3, 5, 2, 1, 5, 3, 6, 7, -2, 3, 5, 5, 55}



One possible test for outliers is to calculate the mean and standard deviation without the presence of the suspected outlier, and then to check whether the suspect piece of data is within three standard deviations of the mean. If not, then it is discarded.

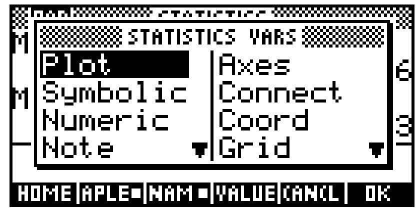


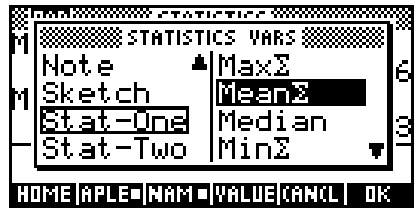
Enter the data without the suspected outlier into column C1 with the calculator in 1VAR mode. Ensure that the **SYMB** view is set up correctly and then press the STATS key.
As you can see on the right, the values of the mean and standard deviation are given in the STATS screen to 12 significant digits.



If we now switch to the **HOME** view, we can recall these values and use them in a calculation to find the upper and lower cut off points for acceptance of data.

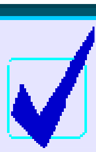
As you can see on the left, the range for acceptance is -3.46 to 10.38, which makes the value of 55 almost certainly an error.





There are two ways to obtain these values. You can type them into the **HOME** view using the **ALPHA** key, or you can use the **VARS** key instead.

If you press **VARS** and then the screen key labeled **APLET** then, assuming the Statistics aplet is active, you will see the view above right. Scroll down to the *Stat-One* variables and then press the right arrow. This puts you in the list (see right) of univariate summary statistics. If you highlight one and press **ENTER** then the variable name will be pasted into the **HOME** view for use. People often find it easier to simply type them. You can obtain the summation sign in **MeanΣ** by using **SHIFT +**.



Calculator Tip

The values of the mean and standard deviation retrieved are those of the *last set calculated*. If you have more than one set of data in the **NUM** view then firstly un**✓CHK** all except the one you want, and secondly press the **STATS** key in order to force a calculation of the values you want. This ensures that the ones you retrieve in the **HOME** view are the ones you want.

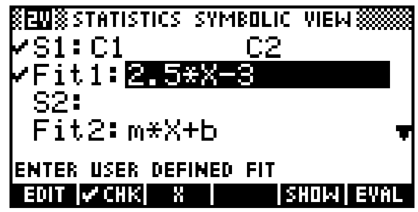
This technique can also be used for bivariate data in exactly the same manner. You should ensure, as per the note above, that only the set of data you want to use is **✓CHK**ed and that **STATS** have been displayed for it.

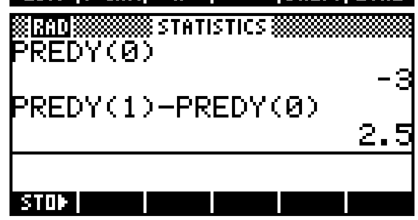
All the usual values are retrievable, with the exception that there is no way to provided to retrieve the coefficients of the line of best fit, since they appear in the **SYMB** screen rather than the **STATS** screen.

The function **PREDY** from **MATH** gives a predicted y value *using the last line of best fit that was calculated*. This means that you must use the **SYMB** view to ensure that your set of data is the only one ✓CHKed and make sure **SYMB SETUP** is set to the correct fit model, and also use the **PLOT** screen and the FIT key to ensure that your set of data was the last one graphed and that it has had its curve of best fit displayed. Until the curve has been displayed, the coefficients are not available or, worse, might belong to another data set.

If you want to gain access to the **PREDY** function algebraically then the simplest way to do this is to enter **F1(X)=PREDY(X)** into the Function aplet, highlight it and press EVAL. The equation will be transferred and can then be analyzed using the normal Function tools.

If you want individual coefficients they can also be obtained from the chosen fit model algebraically. For example, if the line of best fit is $y = m * X + b$ and the fit line is showing in the **SYMB** screen as below then the calculations shown right will give the slope and y-intercept.





$$\begin{aligned} PREDY(0) &= m*0+b \\ &= b \end{aligned}$$

and
$$\begin{aligned} PREDY(1) - PREDY(0) &= (m*1+b)-(m*0+b) \\ &= m+b-b \\ &= m \end{aligned}$$

Finding Fit Coefficients

Linear - **m*X+b** b = **PREDY(0)**
m = **PREDY(1)-PREDY(0)**

Logarithmic - **m*LN(X)+b** b = **PREDY(1)**
m = **PREDY(e)- PREDY(1)**

Exponential - **b*EXP(m*X)** b = **PREDY(0)**
m = **LN(PREDY(1)/PREDY(0))**

Power - **b*X^m** b = **PREDY(1)**
m = **LN(PREDY(e)/PREDY(1))**

Quadratic - **a*X^2+b*X+c** a = **(PREDY(2)-2*PREDY(1)+PREDY(0))/2**
b = **(PREDY(2)+4*PREDY(1)-3*PREDY(0))/2**
c = **PREDY(0)**

or, $$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 4 & 2 & 1 \end{bmatrix}^{-1} \times \begin{bmatrix} PREDY(0) \\ PREDY(1) \\ PREDY(2) \end{bmatrix}$$

Cubic - **a*X^3+b*X^2+cX+d**

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 8 & 4 & 2 & 1 \\ 27 & 9 & 3 & 1 \end{bmatrix}^{-1} \times \begin{bmatrix} PREDY(0) \\ PREDY(1) \\ PREDY(2) \\ PREDY(3) \end{bmatrix}$$

Exponent - **b*EXP(m*X)** b = **PREDY(0)**
m = **PREDY(1)/PREDY(0)**

Trigonometric - **a*SIN(b*X+c)+d**

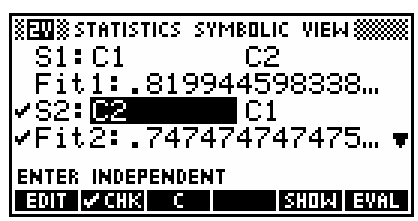
There is no easy way to retrieve the coefficients in the trigonometric equation. The 'simplest' way is to firstly transfer it to the Function aplet by entering **F1(X)=PREDY(X)** into the Function aplet, highlighting it and then pressing **EVAL**. If you now change to the **HOME** view and type **F1(QUOTE(X))** then the equation will appear in the **HOME** view. You can then **COPY** it and edit out any coefficients you want. Clearly this is not ideal. It would be simpler to write them down and re-type them when required!

Correct interpretation of the PREDX function

The **PREDX** function in the **MATH** menu is not really a way of predicting x values despite its name. Algebraically it simply reverses the line of best fit. For example, the equation $\hat{Y} = 0 \cdot 8199X + 1 \cdot 1662$ earlier would use $\hat{X} = \frac{(Y - 1.1662)}{0.8199}$ to predict the X values.

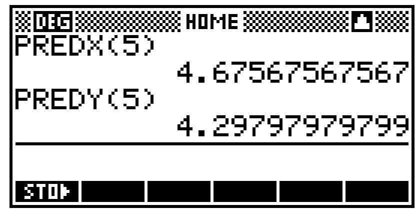
Whether this is mathematically correct depends on how you interpret the **PREDX** function. If, as HP intended, you interpret it to mean "give me an x value which, if used in the **PREDY** function, would give me this y value", then it is correct. However, it should not be interpreted to mean "predict an x value based on this y value" as most people might.

The reason for not using the second interpretation is that the results it gives would then be incorrect. The line of best fit (unlike the correlation) changes as the independent and dependent variables swap roles and can't be simply algebraically reversed in this way. It should not be thought that the hp 39gs & hp40gs are unusual in this odd interpretation. Most calculators' equivalent of the **PREDX** function behave in the same manner.



The formula for the slope b in the line of best fit $\hat{y} = a + bx$ is given by the formula: $b = \frac{S_{xy}}{(S_x)^2}$.

While the value of S_{xy} will not change if the roles of independent and dependent columns are reversed, the value of $(S_x)^2$ on the bottom means that this formula will give a different value if you change which column is regarded as x (independent) and which as y (dependent). This different value for *b* will also mean a different value for *a* and these will not be the values which would result from the simple inverse function.
If you truly wish to predict X values in the sense of the second interpretation, then you should change to the **SYMB** view and enter a new data set **S2** which uses **C2** and **C1** in reversed order, avoiding the need to re-enter the data.



Now use **PREDY** for the second fit line rather than **PREDX**. As you can see in the screen below, the result of **PREDX(5)** with **C1** vs. **C2** is not the same as **PREDY(5)** using **C2** vs. **C1**.

Assigning rank orders to sets of data

It is occasionally handy to be able to assign rank orders to a set of data. You might be running a Quiz Competition Night, or recording times for the 100 meter sprint, but in either case it is handy to be able to sort the data into descending order and assign rankings. This is easy for small sets of data, but becomes difficult for larger sets.

Competitor	Time
1	12.23
2	11.47
3	11.34
4	12.87
5	12.23
6	11.30
7	10.51
8	11.34
9	11.46
10	12.34
11	12.23
12	11.50
13	12.01
14	11.97
15	12.05
16	12.87
17	12.02
18	12.52
19	11.37
20	10.75

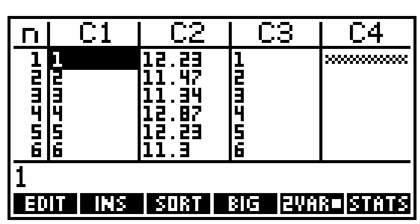
Let us assume a set of 20 competitors in the 100 meter sprint, with times recorded to two decimal places, and competitors numbered 1 to 20. Suppose that the results were as shown left.

Enter the competitor numbers as column **C1** and the times in column **C2**.

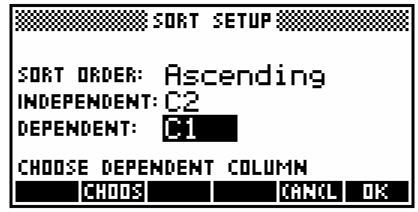
I'm going to assume here that the competitor numbers also run from 1 to 20 but this may not be the case. In addition to this, put the numbers 1 to 20 into column **C3** also. They can be a bit tedious to type in for lists longer than 20, so you could use the expressions **MAKELIST(X,X,1,20,1) ▶C1** and **C1▶C3** to shortcut the process, replacing 20 with whatever is needed.

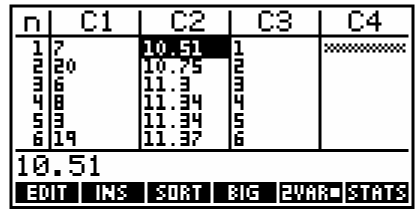
Make sure that the **2VAR■** option is selected.

The result should look like this…



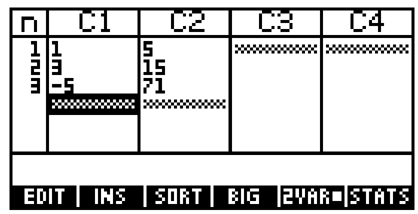
Now position the highlight on column **C2** and press the **SORT** key. In the **SORT SETUP** screen (shown right) enter **C1** as the *Dependent* column. This will have the effect of pairing columns **C1** and **C2** and then sorting column **C2** into ascending order, re-arranging column **C1** to retain the existing data pairings. When ready, press **OK**. The results of this sort are shown right.





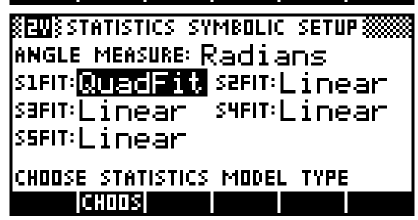
The final column **C3** has not been re-arranged. With a few alterations, it will contain the rankings. Looking down the column it can be seen that there are two values of 11.34 (the 4th and 5th). Their ranking should therefore both be changed to 4. Continuing this process down the length of column 3 will produce the rankings for the whole 20 competitors. For example, competitor number 7 has a time of 10.51 and a rank of 1.

Using Stats to find equations from point data

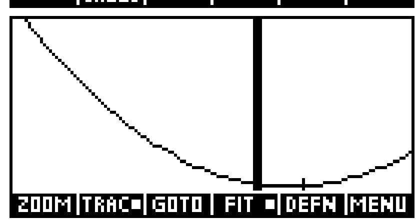


eg. 1 *Find the equation of the quadratic which passes through the points (1,5), (3,15) and (-5,71).*

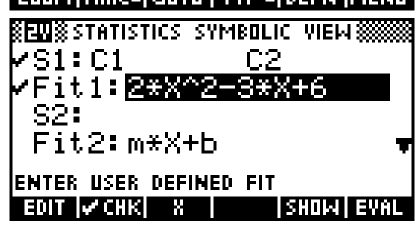
In the Statistics aplet, choose **2VAR** mode and enter the data. Now change to the **SYMB SETUP** view and choose the *Quadratic* data model. Clearly this process will only work for those equations which are available as data models but that does offer quite a few choices (see page 126).



Change to the **PLOT** view using the **VIEWS** *Auto Scale* option and press the **FIT** key. Don't worry that the scale is not good because we don't care about the graph. It only needs to be drawn in order to calculate the fit equation.



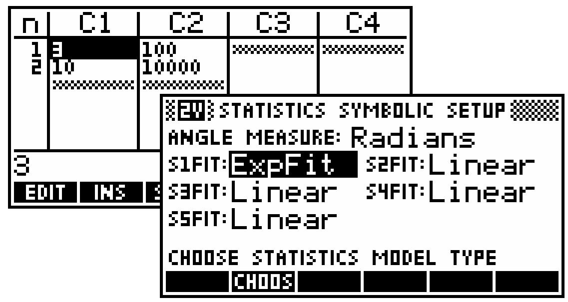
Finally, change back to the **SYMB** view and see the equation, pressing **SHOW** if necessary.



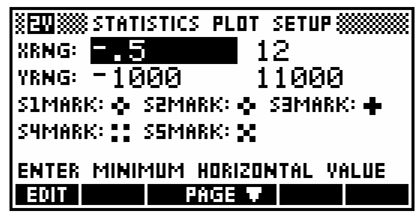
eg. 2 *A population of bacteria is known to follow a growth pattern governed by the equation* $N = N_0 e^{kt}$ $;t \geq 0$. *It is observed that at t = 3 hours, there are 100 colonies of bacteria and also that at t = 10 hours there are 10 000 colonies.*

i. *Find the values of* N_0 *and of k.*
ii. *Predict the number of bacteria colonies after 15 hours.*
iii. *How long does it take for the number of colonies to double?*

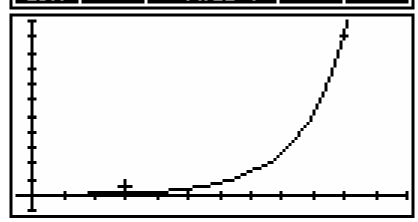
i. Find N_0 and *k*.



Start up the Statistics aplet, set it to **2VAR** and enter the data given. Change to the **SYMB SETUP** view and specify an *Exponential* model for the data.



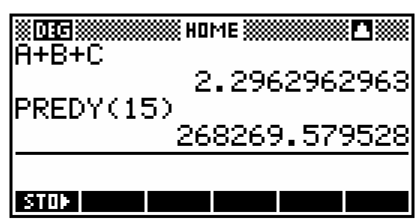
Either use the **VIEWS** *Auto Scale* option, or change to the **PLOT SETUP** view and adjust it so that it will display the data. This is not really needed, since the line of best fit is what we need and it will be calculated even if the data doesn't show.



Now change to the **PLOT** view and press **FIT**. Wait while the line draws.

Change to the **SYMB** view, move the highlight to the equation of the regression line and press **SHOW**. Rounded to 4 decimal places, this gives an equation of $N = 13{\cdot}8950\,e^{0{\cdot}6579\,t}$.

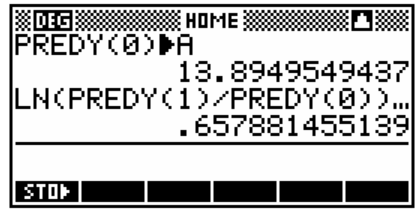
ii. Predict N for t = 15 hours.



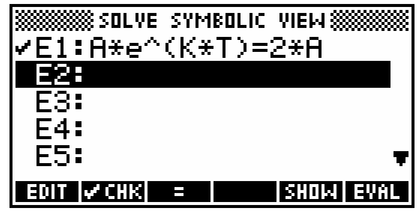
Change to the **HOME** view and use the **PREDY** function or use the facilities in the **PLOT** view.

Result: 268 269 colonies.

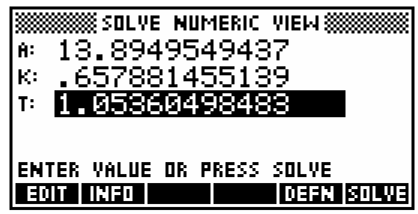
(iii) Find t so that $N = 2N_0$.



The value of N_0 is the y intercept of the line of best fit. These values from the curve of best fit are not directly accessible but can be retrieved using the **PREDY** function (see page 135). This is shown in the screen shown right. Store the results into memories **A** and **K**. This saves having to re-type them from the **SHOW** screen.



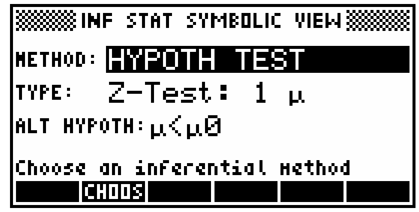
Now switch to the Solve aplet and enter the equation to be solved. Changing into the **NUM** view, you should find the values of A and K already defined (that was why we stored the curve values into the appropriate memories), so move the highlight to T and press **SOLVE**.



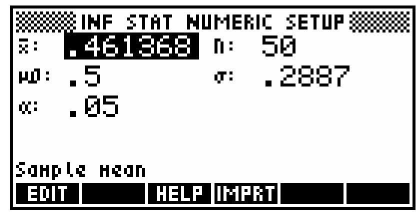
Result: Doubling time is 1.0536 hours.

Note: An alternative to this would be to only retrieve the value of N_0 and store it into **A**. The Solve equation could then be **PREDY(T)=2*A** and the result would be the same.

THE INFERENCE APLET

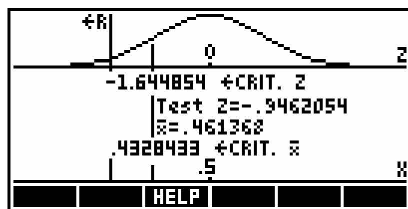


This aplet is a very flexible tool for users investigating inference problems. It provides critical values for hypothesis testing and confidence intervals, and does this not only quickly but in a visually helpful format. It will be assumed in the explanations that follow that the reader is familiar with the concepts of hypothesis testing.

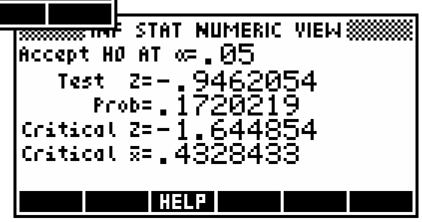


In the Inference aplet the **SYMB** view is used to choose the test to be applied.

The **NUM SETUP** view is used to enter the information required to apply the test. These results can either be entered by hand or imported from the Statistics aplet using the **IMPRT** button.

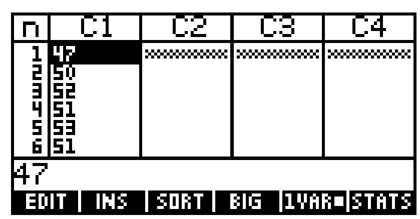


The results are displayed either numerically in the **NUM** view or graphically in the **PLOT** view.



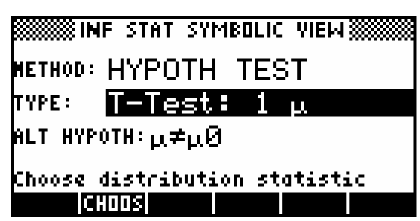
Generally the information that is required for the **NUM SETUP** view is obtained by importing it from the Statistics aplet. The logic is that there is little point in including facilities in the Inference aplet that are already more than ably supplied in the Statistics aplet.

Hypothesis test: T-Test 1-μ



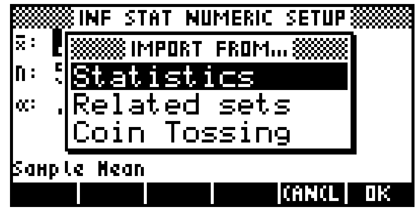
> *A company makes boxes of matches which are supposed to contain an average of 50 matches. A student has counted the contents of 20 boxes and found the results below. Using a 5% confidence level, decide whether the claim is correct.*
>
> *47, 50, 52, 51, 53, 51, 49, 49, 52, 53, 54, 48, 53, 55, 52, 54, 54, 53, 48, 48*

Enter the data into the Statistics aplet then change to the Inference aplet. Choose the test and the alternate hypothesis in the **SYMB** view of the Inference aplet.

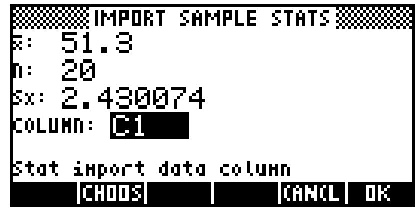


In this case we are working with a single sample and we do not know the standard deviation of the underlying population, so we will use the Student-t test and the alternate hypothesis that the mean of the real underlying population from which the sample was drawn is not equal to that of the proposed underlying population.

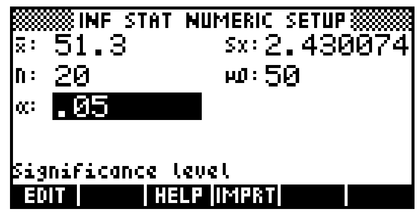
Change now to the **NUM SETUP** view to enter the required values. Rather than entering them by hand, press the IMPRT key. If you have more than one copy of the Statistics aplet (under other names) then you will be presented with a list of aplets from which to choose.



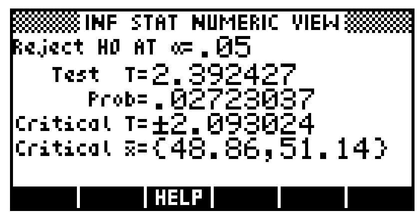
Once you have chosen the aplet, you need to nominate the column from which to import the data. The default is column **C1**, which is what we want in this case, but you can press the CHOOS key to select from a list of any other columns which contain data. When you have the correct column, press OK.

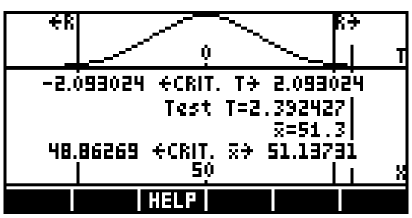


Enter the population mean μ_0 = 50, and check that the test level is correct at 0.05 (5%).



If you now change to the **NUM** view you will see the inferential data in numeric form. The test Student-t value is given as 2.392 and the probability of obtaining such a value as 0.0272. The critical t values for this test level of 5% are given as ± 2.093 and the critical boundaries for the sample mean as 48.86 and 51.14.





The information can be seen in a more visual form if you change to the **PLOT** view as can be seen left. The upper normal curve shows the critical range for the t values, together with a short vertical line showing the position of the Test T value. The lower horizontal line shows the equivalent critical sample mean range.

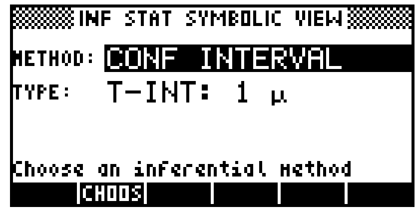
The test value for the Student-t and the sample mean are also listed in the middle of the screen. The regions for rejection of the null hypotheses are shown at the very top of the screen by the '←R' and 'R→'.

We assume, by statistical theory, that the distance $\left(\bar{x}-\mu\right)$ is normally distributed. If the null hypothesis is true then the mean of this new distribution should be zero. Our result is indicating that our particular value of $\left(\bar{x}-\mu\right)$ is of such a size that if the null hypothesis is true then the probability of it appearing by chance is less than 5%.

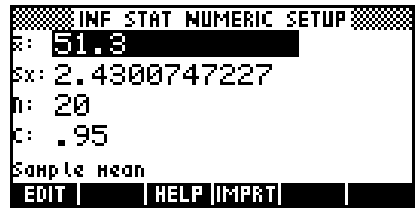
In this case, whether you work from the **PLOT** view or the **NUM** view, it is clear from the evidence that the null hypothesis should be rejected, with less than a 5% chance of this rejection being incorrect. The alternate hypothesis that the mean is not 50 should be accepted. Of course in this case, despite some boxes containing less than 50 matches, the actual mean seems to be above 50 so we can hardly condemn the company for putting more matches in the boxes than they need to!

Confidence interval: T-Int 1-μ

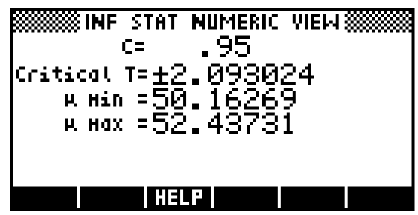
In the previous example we found that the evidence of our sample indicated that the mean number of matches in the boxes was not 50. Suppose we now want to know, at the 95% confidence level, within what range of values the true population mean lies.



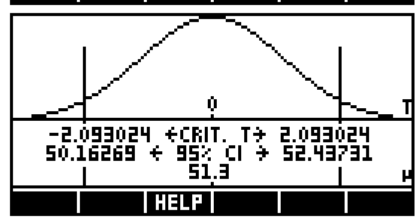
Change back to the **SYMB** view and **CHOOS** the method of *Conf Interval*.
The type of interval is converted to the equivalent type of *T-INT: 1 μ*.



In the **NUM SETUP** view use the import facility as before to import the values from our sample data. The default confidence level is 99% so you will need to change that to 0.95.



Changing to the **NUM** view gives the minimum and maximum values for the population mean of 50.16 to 52.44 at a 95% confidence level.



As before, a more visual display can be seen in the **PLOT** view.

Thus the sample data indicates in our two examples that:

- we can be confident the average number of matches is not 50 with less than a 5% chance of being wrong, and
- we can conclude, with a confidence of 95%, that the true average number of matches is between 50.16 and 52.44.

Hypothesis test: T-Test μ_1-μ_2

A farmer compared the 15-day mean weight of two sets of chicks, one group receiving feed supplement A and the other supplement C. Twenty two chicks only one day old were assigned randomly to the two groups. To distinguish between the two groups of eleven, which were caged together to minimize other influences, the heads of the chicks were stained red and purple respectively with a harmless vegetable dye.

The individual weights were recorded in the table below. From the means at the end of the table it appears that supplement A produces a higher mean weight.

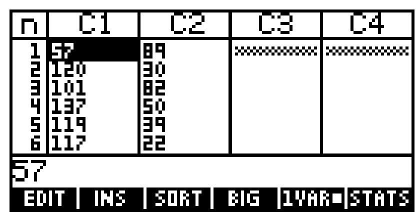
	Weights											$\bar{x}$	σ_X
Supp. A	57	120	101	137	119	117	104	73	53	68	118	97	27.75
Supp. C	89	30	82	50	39	22	57	32	96	31	88	56	26.54

The farmer wishes to know, at the 1% test level, whether this is statistically significant.

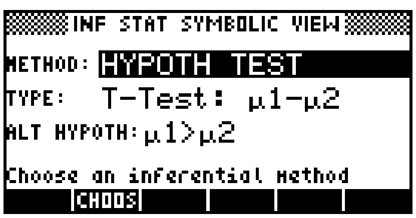
Our hypotheses are:

H_0: The effects are the same $(\mu_1 = \mu_2)$ ie. There is no effect from the supplement.

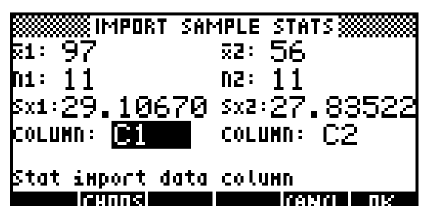
H_A: Supplement A is better $(\mu_1 > \mu_2)$



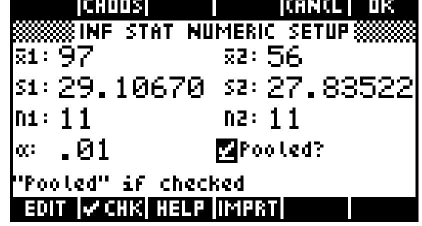
Enter the data into columns **C1** and **C2** of the Statistics aplet.



We are dealing with two independent samples in this case and so we need to choose from those tests which involve two samples.

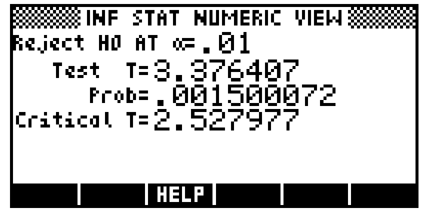


Since we know the standard deviations of only these samples, we must again use a Student-t test. We then change to the **NUM SETUP** view to import the summary statistics. The import screen can be seen on the right.

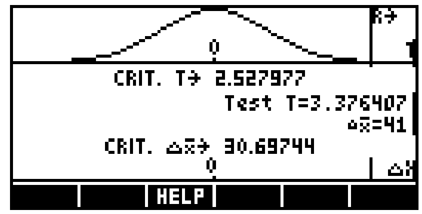


The results of the importation can be seen right. Since the two standard deviations for the samples are similar, and the sample sizes are the same, a more accurate estimate of the population standard deviation will be obtained if the data is pooled, so ensure that this option is ✔CHKed.

The **NUM** view shows the critical values. We can see that the probability of obtaining a test student-t value of 3.38 is 0.0015 and this is well below the permitted test level of 1%.



The **PLOT** view also shows that the vertical line representing the value of $\bar{x}_1 - \bar{x}_2$ or $\Delta\bar{x}$ is well into the region of rejection indicated by the R→.



From the evidence we should reject the null hypothesis and accept instead that the average weight of the group receiving supplement A is significantly higher than those receiving supplement C with only a 1% chance of being incorrect.

Hypothesis test: Z-Test 1-μ

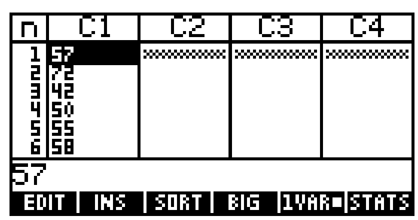
A teacher has developed a new teaching technique for hearing-impaired students which he believes is producing significantly better results. He wishes to publish a paper on this and needs to check his results statistically.

A standardized test is available for which it is known that the normal performance of hearing-impaired students at the same stage of study has a mean of 53.6% and a standard deviation of 12.2%. When he applies this test to his class of 23 students their scores are shown below. The teacher believes that this data shows that his students are scoring significantly better and wishes to test this at a level of 5%.

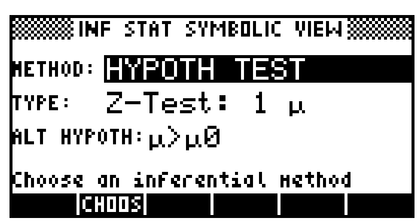
Test results:

{57, 72, 42, 50, 55, 58, 59, 38, 45,53, 77,57, 52, 69, 50, 55, 59, 68, 62, 63, 53, 56}

As usual we begin by entering the data into column **C1** of the Statistics aplet.



Changing to the Inference aplet, we choose a Hypothesis test using *Z-test: 1 μ*, since we know the population standard deviation.

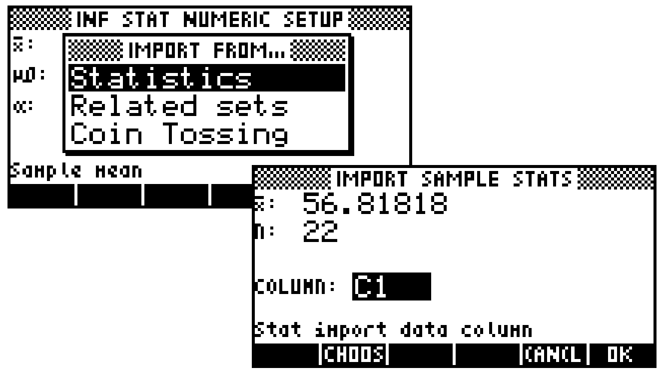


The hypotheses are:

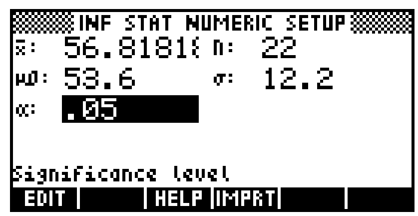
H$_0$: The sample is drawn from a population whose mean is the same as the standardized population $(\mu = \mu_0)$.

H$_A$: The sample is drawn from a population whose mean is larger than that of the standardized population $(\mu > \mu_0)$.

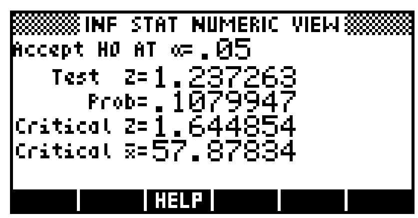
Change to the **NUM SETUP** view, you can use the import facility to import the summary statistics from the Statistics aplet.



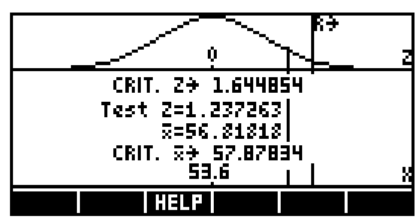
Enter the values for the mean and standard deviation of the standardized test, and the significance level of 0.05 (5%).



If we now change to the **NUM** view we can see that the test z score is less than the required critical z*, and the probability of obtaining a mean of the value found is 0.1080, which is larger than the required test value of 0.05.



In the **PLOT** view, we can see visually that the vertical line representing the sample mean is not within the region of rejection marked by the R→.



From the evidence the teacher must reject the alternate hypothesis and conclude that it is not possible to say at the 5% level of significance that his class has averaged significantly higher than the standardized population from which the test was drawn. He should re-think his proposed paper or his new teaching method. Alternatively, from the diagram in the **PLOT** view it seems that his mean is not far from being significant. Perhaps he simply needs to collect more data in the hopes that this may back up his view. The result he has obtained is, after all, only a probability and further investigation may give a different view.

THE EXPERT: CHI^2 TESTS & FREQUENCY TABLES

We will start with a small digression to look at a simple inferential problem which can be solved using only the Statistics and Solve aplets.

Using the Chi^2 test on a frequency table

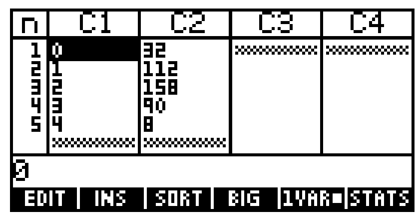
> *"Four coins are tossed 400 times and the number of heads noted for each toss. The results are shown below. Using the Chi^2 test at a 5% level of confidence, indicate whether the coins may be biased."*

Number of heads	0	1	2	3	4
Frequency	32	112	158	90	8

We would expect that for an un-biased set of coins the distribution would be binomial. Our hypotheses are:

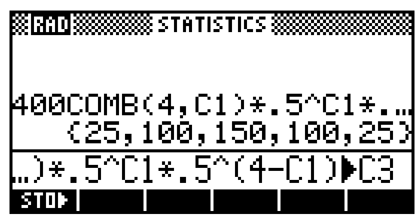
H_0: The number of heads is binomially distributed (n=4 & p=0.5)
H_A: The number of heads is *not* binomially distributed (n=4 & p=0.5)

Begin by entering the data into the first two columns of the Statistics aplet (right).



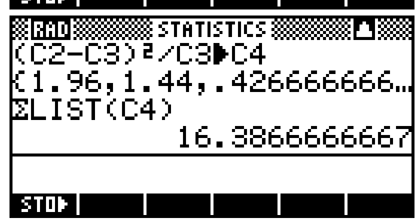
We now need to calculate the expected values based on our null hypothesis. The expected values are based on the assumption that the results are binomially distributed with n=4 and p=0.5.

We can do this in the **HOME** view using the calculation shown right (and below), inserting the results into column **C3** using the **STO▸** button.

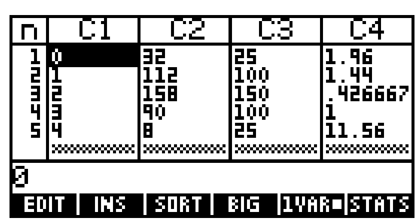


400*COMB(4,C1)*.5^C1*.5^(4-C1)▸C3

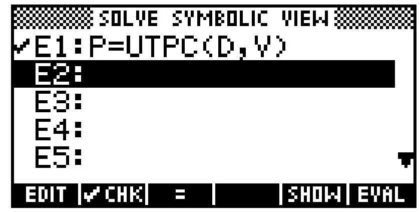
We can now calculate the X^2 value as the sum of the values in column **C4** using the **∑LIST** function. The calculations are shown right, placing the individual values into column **C4** for inspection if required and then finding the sum of the column.



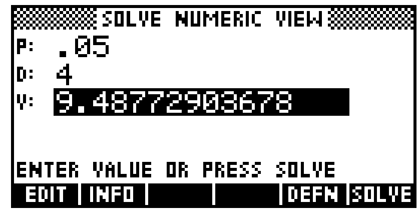
The values can be seen by changing to the **NUM** view.



In the **MATH** menu, *Probability* section (see page 208), there is a function called **UTPC** (Upper-Tailed Probability Chi-squared) which will give the critical X^2 probability for a supplied number of degrees of freedom and a value. In this case we would like the value for a given probability so we will enter the formula into the Solve aplet.



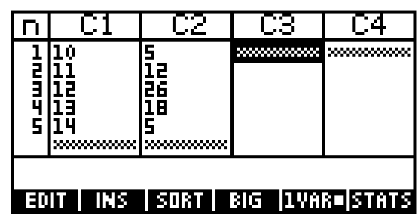
Change to the **NUM** view, enter the known parameters of **D=4** and **P=0.05**, and **SOLVE** for the critical X^2 value.



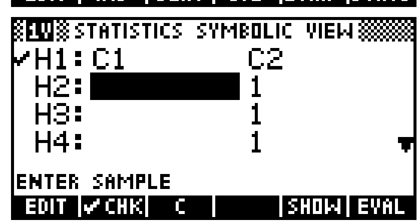
Since our value of 16.387 is larger than the critical value of 9.488 we conclude that we must reject the null hypothesis and judge that the observed values do not follow this binomial distribution and hence that the coin is probably biased. This does not say that it is not binomially distributed; just not with those parameters.

Importing from a frequency table

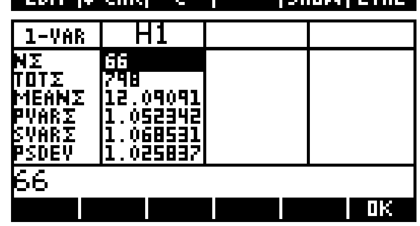
The import (**IMPRT**) facility of the Inference aplet has a small weakness in that it can't import from paired columns defining a frequency table.



For example, suppose we use columns **C1** and **C2** to define a frequency table, ensuring that it is registered in the **SYMB** view as shown right.

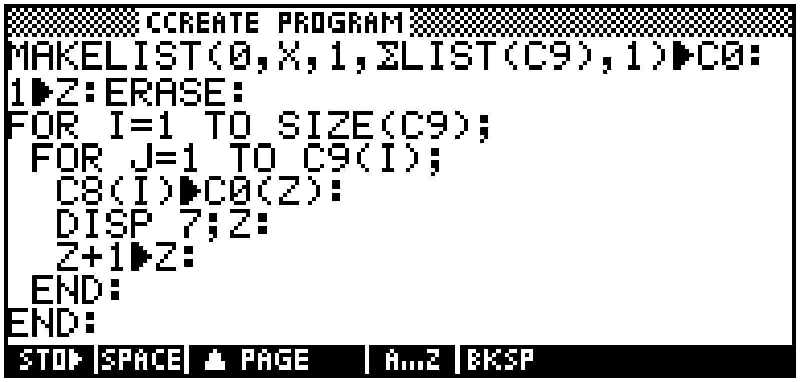


If you now change to the **NUM SETUP** view of the Inference aplet and try to use the import facility you will find that you can import the mean and standard deviation from either column **C1** or **C2** but not from both at once.

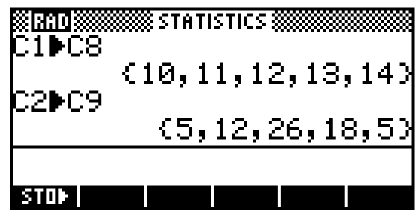


What's needed, of course, is a way to expand the paired frequency table columns into a single column listing all values. You can, of course, create the expanded columns by hand but the program on the next page will do this for you.

To create it, go to the **Program Catalog** view and press the **NEW** key. Enter any name you want, such as 'CCreate'. Now type in the code below. The program is set up to take a frequency table defined in columns **C8** and **C9** and convert them into a single column stored in column **C0**. The program uses columns 8, 9 & 10 because there is seldom data in them.

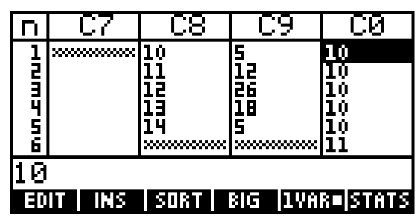


We'll now use this program to expand the frequency table I built earlier. The first thing we need to do is to move columns **C1** and **C2** to **C8** and **C9**. Rather than re-typing, this can be done in the **HOME** view as shown.



Now change into the **Program Catalogue** and **RUN** the program you created. Assuming that it has no errors you will see a running count as it creates the new column. This is just to give you something to watch while it works.

If you then change to the **NUM** view you will find that column **C0** contains the result.



You can now import that column's mean and standard deviation into the Inference aplet. After you have used the column you will probably also want to delete columns **C8, C9** and **C0** to save space.

Bear in mind that if you use this program to create a column containing hundreds or even thousands of values then the program will take a *long* time to complete. In the case of thousands of values you may even exceed the calculator's memory in the **MAKELIST** command. It might be easier to use the Statistics aplet to calculate the required mean and standard deviation in the normal way and then enter the values into the Inference aplet either by hand or by using the **VARS** menu.

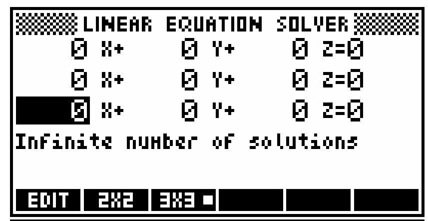
The Linear Solver Aplet

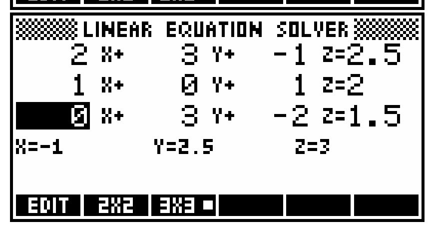
This is a very easy aplet to use. It is designed to solve simultaneous linear equations in 2 or 3 unknowns. If there is a solution it will display it. Otherwise it will indicate whether there is no solution or an infinite number of solutions.

Example 1

Solve the system of equations:

$$\left.\begin{aligned} 2x+3y-z&=2.5\\ x+z&=1.5\\ 3y-2z&=1.5 \end{aligned}\right\}$$





In the Aplet view, **START** the aplet and you will see the view shown right.

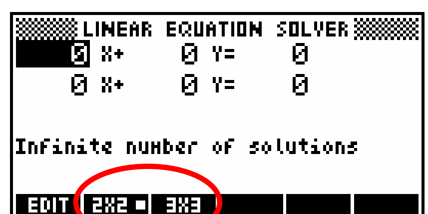
Enter the coefficients from the equations, including zeros for any missing coefficients. Negative coefficients must be entered using the (-) button rather than using 'subtract'.

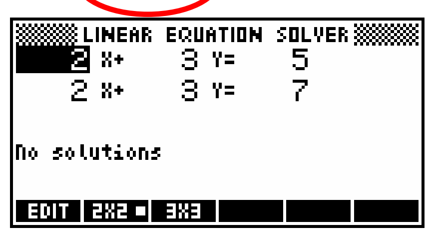
As you do so the solution will be displayed at the bottom of the screen. In this case there is a unique solution of *x = -1*, *y = 2.5* and *z = 3*.

Example 2

Solve the system of equations:

$$\left.\begin{aligned} 2x+3y&=5\\ 2x+3y&=7 \end{aligned}\right\}$$





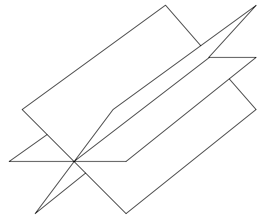
As you may be able to see, this is a pair of parallel lines and so has no solution. To see this on the calculator we must first change to the 2x2 view, meaning 2 equations in 2 unknowns. At the bottom of the screen you will see buttons labeled **2X2** and **3X3**. Currently the 3x3 button should be selected, as shown by the dot next to it on the label: **3X3 ■**. Click on the **2X2** button and it will change into the 2x2 view as shown above right.

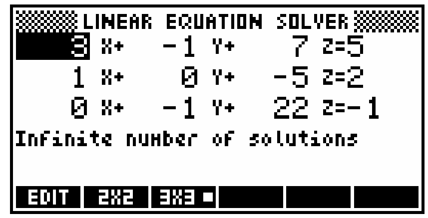
Enter the coefficients for the equations above. As you can see in the screenshot to the right, the result is shown as "No solutions" as expected.

Example 3

Solve the system of equations:

$$\left.\begin{aligned} 3x - y + 7z &= 5 \\ x - 5z &= 2 \\ -y + 22z &= -1 \end{aligned}\right\}$$

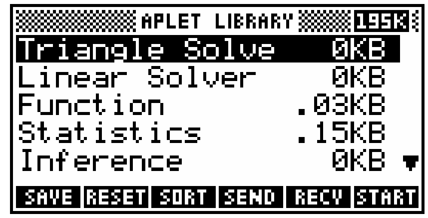




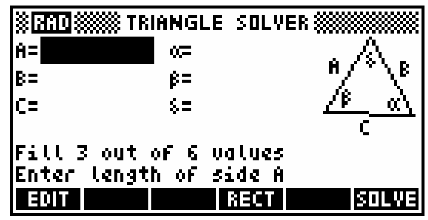
Although it may not be obvious at first glance, this system of equations corresponds to a 'spindle' of planes in 3-space as shown in the diagram above right. This situation allows infinite solutions anywhere along the line of intersection of the three planes.

As you can see right, the calculator has correctly indicated the situation.

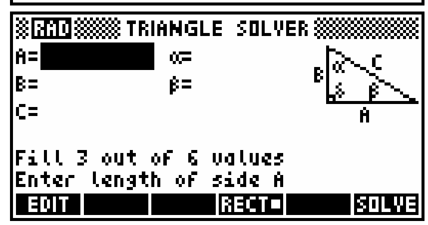
THE TRIANGLE SOLVE APLET



This aplet allows you to solve for missing sides and angles in a triangle, either right angled or not. Unlike most aplets it does not have a **SYMB**, **NUM** and **PLOT** view but only the dual view discussed below.

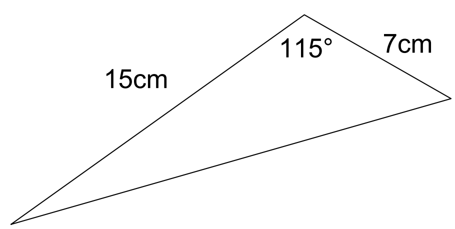


When you first start the aplet you will be presented with one of two views. The view right is for a general triangle, while the second view assumes a right triangle. Obviously the amount of information which must be supplied is less for the right triangle.



The choice of which view you use is controlled by the button labeled RECT, short for “rectangular”. When RECT■ is selected (shown by the dot) you are supplying information for a right triangle.

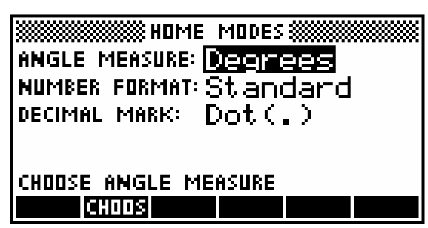
In both cases the procedure is to supply sufficient information to allow the triangle to be solved, hence the message to “Fill 3 out of 6 values”. This request should be taken with a pinch of salt. For example, in the case of a right triangle it is actually only necessary to supply 2 values despite what is stated.



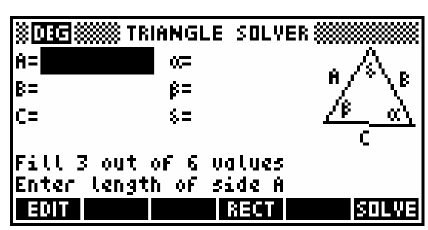
Example 1

Solve the triangle shown right.

Use the Aplet view to select and START the aplet.

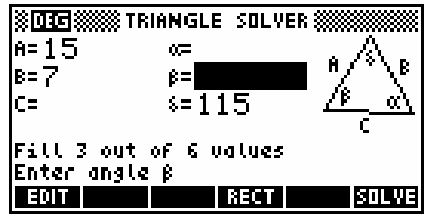


The second is to ensure that we are working in degree mode. Change to the **MODES** view and choose **Degrees**.



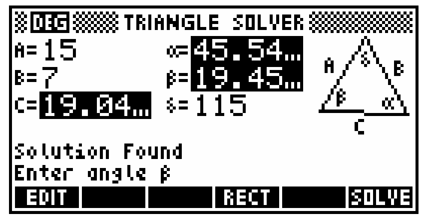
Now press **SYMB** (or **NUM** or **PLOT**) to change back to the view shown right.

Since this is not a right triangle, the first step is to ensure that **RECT** is not selected, as is shown right. Any of the three angles α, β or δ can be used to represent the 115° angle.



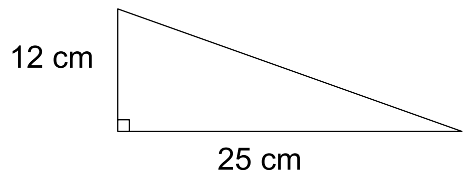
In this case I will use δ for no other reason than that it is at the top of the illustration, just as it is in the diagram of the triangle. This means that the 15 cm goes into the **A** field and the 7 cm into the **B** field. Enter those values, using the arrow keys to move from one to another. The result should be the screen shown right.

Pressing the button labeled **SOLVE** will result in the remaining 3 values being filled in, as shown.



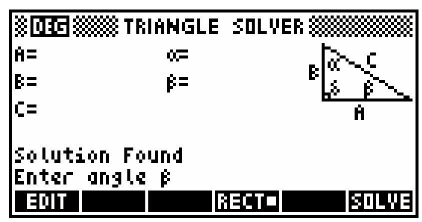
The calculated values are highlighted for convenience.

Example 2



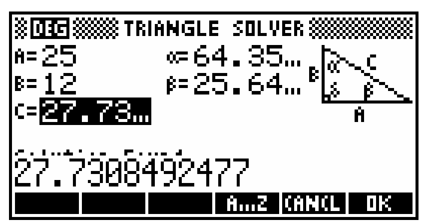
Find the length of the hypotenuse for the triangle shown right.

Since we don't want the sizes of the angles it doesn't really matter what angle mode the aplet is set to. If you worked the previous example then it is probably still in degree mode.



Press the **RECT** button to change the screen into the right triangle format as shown right. Pressing **SHIFT CLEAR** will remove the remaining values from the previous problem.

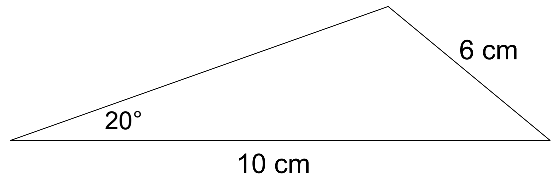
Use the arrow keys to move to the **A** and **B** fields and enter the values of 25 cm and 12 cm respectively. Then press **SOLVE**.



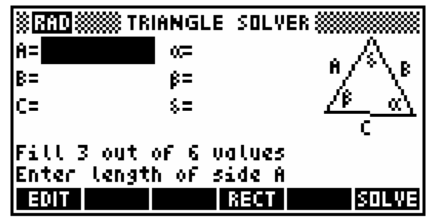
The result should be as shown right, giving a length for the hypotenuse of 27.73 cm. In the example screen I have also pressed the **EDIT** button (with the highlight on **C**) so that I can see more than 2 decimal places.

Example 3

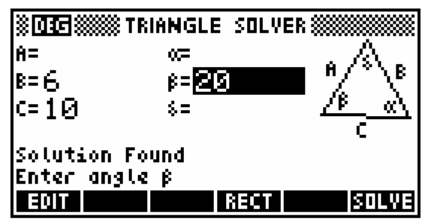
Solve the triangle shown right.



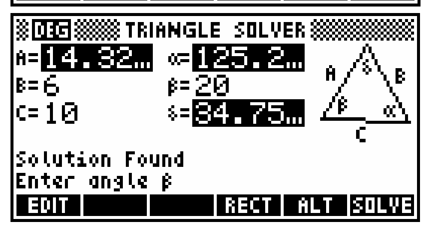
This is an example of a triangle that has two possible solutions, generally referred to as "The Ambiguous Case". The calculator will give both possible solutions.



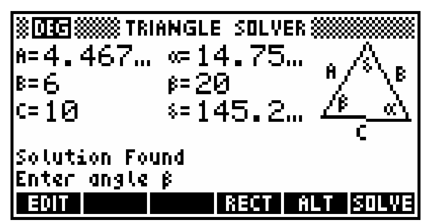
Begin by setting the calculator into **Degree** mode, if it is not already. Change into the **SYMB** view and ensure that the non-right triangle is selected as shown.



Purely to maintain orientation, we will select **C** as the side that is 10cm long and enter the values shown right. Notice that as soon as sufficient information has been entered the message "Solution Found" appears.



Press SOLVE and the calculator will fill in the missing values. The additional button label of ALT will appear to announce that an alternate solution is available.



Pressing ALT will alternate repeatedly between the two solutions.

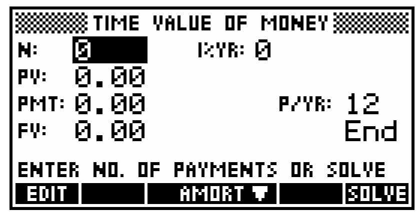
26

THE FINANCE APLET

This aplet is designed to allow users to solve time-value-of-money (TVM) and amortization style problems quickly and easily, as well as ordinary compound interest problems.

Compound interest problems involve bank accounts, mortgages and similar situations where “money earns money”. TVM problems involve the use of the idea that the value of money changes with time - a dollar today is worth more than the same dollar some years from now. For example, that a dollar invested today can generate more money than the same dollar invested later.

The calculator manual contains a lengthier explanation including cash flow diagrams for those who need it, as does any high school or college textbook.



When you **START** the aplet you will see the initial view shown right.

Pressing **SYMB**, **NUM** or **PLOT** will make no difference to this aplet as it is quite limited and only has the one view, consisting of two related pages.

Parameters

There are a number of parameters or variables which must be either supplied or solved for. These are:

N - The total number of compounding payments or payments.

Mode - This has the value of **Beg**(inning) or **End** depending on when payments occur relative to the compounding periods - at the beginning or the end.

P/YR - The number of payments per year.

I%YR - The nominal interest rate or investment rate per year. This is then divided by **P/YR** to find the nominal interest per compounding period. This is the rate actually used in the internal calculations.

PV - This is the present value of the initial flow of cash. In a loan, this is the amount of the loan. In an investment, the amount invested. **PV** is always the amount at the start of the first period, however long that may be.

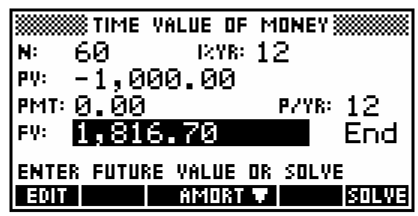
PMT - This is the size of the periodic payment. The assumptions made are that all payments are the same size and that no payments will be skipped. Payments can occur at the beginning or the end of a compounding period, depending on the setting of **Mode**.

FV - The future value of the investment or loan. This could be the amount in a bank account after a period of years, the residual on a lease, the amount still owing on a loan after **N** repayments, or the remaining value of an investment which has been paying income as an annuity.

When using the aplet is it important to visualize the cash flow in terms of positive cash (inwards) or negative cash (outwards). This can be illustrated simply via the following example. Always bear this principle in mind when deciding how to plan your setup of the aplet.

Ordinary compound interest

Invest $1000 into a bank account paying 6.5% per annum for 5 years. Interest is calculated monthly and credited to the account at the end of each month. What is the investment worth at the end of the period?

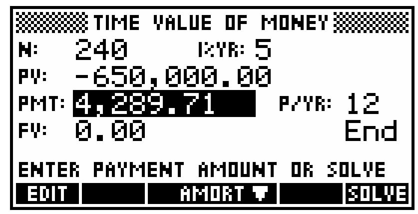


Looked at from the point of view of the person investing the money, the cash flow is initially outwards (negative) by $1000. The final payment is inwards (positive) and no payments (withdrawals) are made during the period of the investment. Five years of monthly payments means that **N** is 60. The view on the right shows the problem on the calculator. The **SOLVE** button has been pressed to give a future value **FV** of $1816.70.

Calculator Tip
As can be seen above, the designers of this aplet chose to display money to 2 decimal places and using comma separators. This choice is independent of the settings in the **MODES** view. When entering values do not use commas as they will simply result in a syntax error.

Annuities

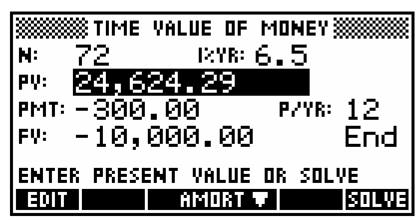
An engineer retires with $650,000 available for investment. She invests the money in a portfolio which is expected to have an average return of 5% per annum. She wants to have the account pay a monthly income to her and asks the accountant to assume that the income must last for 20 years. What income can be withdrawn?



The **PV** for this problem is negative because, from the point of view of the engineer, the money flow is outward from her to the investment portfolio. The value of the regular payment, which has been solved for on the right, is positive as it is being paid from the portfolio to her. We also assume that the future value **FV** is to be zero since the income is only to last for 20 years. The value of **N** is 240, representing 20 years of monthly payments. On this basis the monthly annuity can be $4289.71

Loan calculations

You wish to purchase a car by taking out a loan. The current interest rate is 6.5% and you can afford to make monthly payments of $300. You want to take out the loan over a period of 6 years and to still owe $10,000 at the end of that period (you expect an investment to mature at that time to pay the final amount). How much can you borrow?



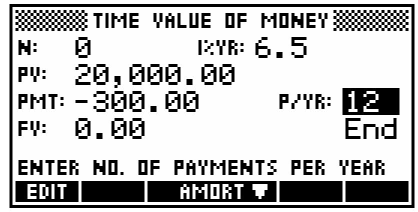
In this case, from your point of view, the payments and the **FV** final residual are outgoing (negative) since they are made to the bank. The present value (the loan) is positive. **N** is set to 6 years of monthly payments.

Press **SOLVE** to find the **PV** and this is the amount that can be borrowed. It will be positive since it is going from the bank to you. In this case it is clear that you can afford to borrow $24,624.29.

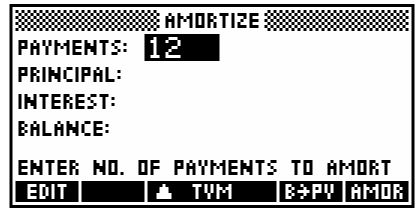
Amortization

The second page of this aplet allows amortization calculations in order to determine the amounts applied towards the principal and interest in a payment or series of payments.

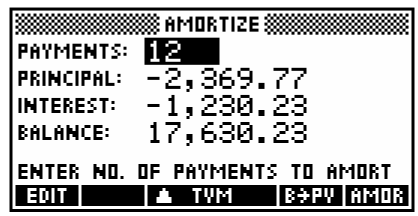
Suppose we borrow $20,000 at an interest rate of 6.5% and make monthly payments of $300. The initial situation is as shown in the screen on the right.



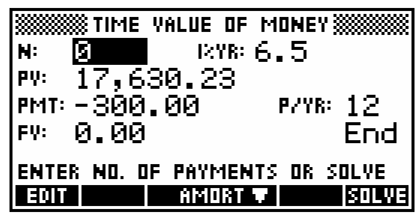
Press the **AMORT ▼** button to change to the amortization screen. The initial appearance is as shown. As can be seen, the default number of payments to amortize over is 12.



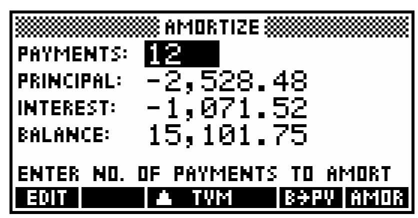
Pressing the **AMOR** button will amortized the loan over the first year. As can be seen right, $2,369.77 has been paid off the principal, leaving a balance of $17,630.23. Of the payments made, $1,230.23 went towards the payment of interest.



Pressing the **B→PV** button will transfer the balance back to the previous page as the new value of the **PV**.



At this stage you can amortize again for another 12 months as shown right. This can be continued indefinitely.

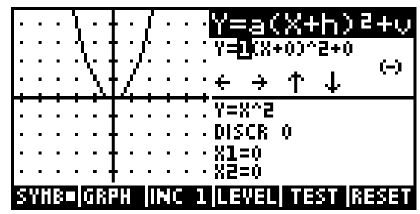


THE QUAD EXPLORER TEACHING APLET

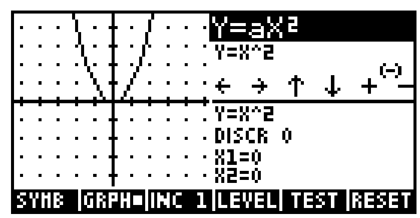
Rather than being a multi-purpose aplet, this is a teaching aplet specialized to the single use of exploring graphs of quadratics. As such it does not have the normal **SYMB**, **NUM** and **PLOT** views, but only a single multi-purpose view.

Objectives

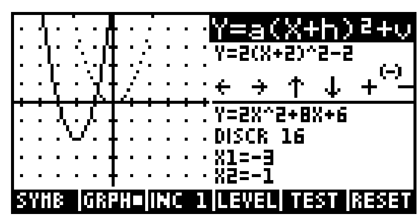
Using the Quadratic Explorer aplet, the student will investigate the behavior of the graph of $y = a(x+h)^2 + v$ as the values of *a*, *h* and *v* change. This can be done both by manipulating the equation and seeing the change in the graph, *and* by manipulating the graph and seeing the change in the equation.



Press **START** to see the main screen of the aplet, shown right. This aplet has only the one screen, rather than the usual **PLOT**, **SYMB** and **NUM** views. If the aplet has been used before then it may be necessary to press the **RESET** screen key to go back to the opening view.



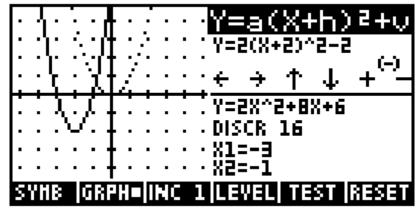
The first choice made by the user should be the **LEVEL**. The default level is that of Y=a(X+h)^2+v , which allows the student to change all parameters at once. By having the student choose the levels of Y=aX^2, Y=(x+h)^2 or Y=X^2+v, the teacher can confine the study to the effects of only one coefficient at a time if desired.



The default state for the aplet is to be in **GRPH** mode. In this mode the student uses the keys listed below to control the appearance of the graph, with these changes reflected automatically in the two equations on the right half of the screen. The original y=x^2 graph is kept on the screen (dotted) for comparison and a grid is supplied to allow the user to see movement more clearly.

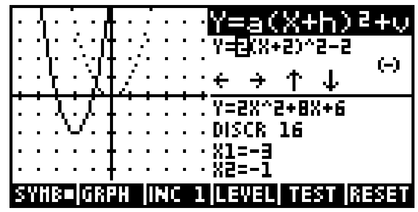
The control keys (listed onscreen) are:

▲ ▼ ◄ ►	move the graph in the obvious way.
+ and −	cycle the value of ‘a’ in y=a(x+h)^2+v through the values 0.1, 0.2, 0.3, 0.5, 1, 2 & 5. No other values are possible.
(-)	(the ‘negative’ key) changes the sign on the ‘a’ coefficient turning the graph upside down.



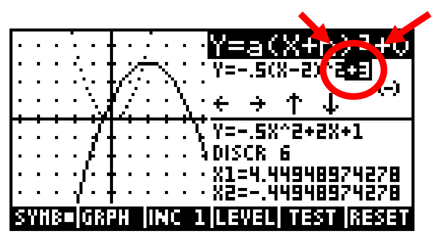
As can be seen in the screen shots right, the bottom half of the screen shows the roots (if any), the value of the discriminant and the equation in the form $y=ax^2+bx+c$.

The key labeled **INC 1** changes the 'step size' of the movements on the screen. Possible values for the increment are 0.5, 1 and 2.

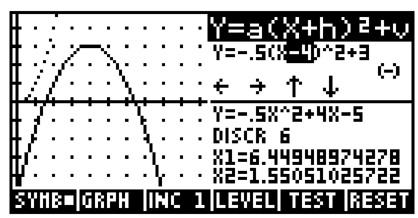


Pressing **SYMB** on the calculator, or the screen key labeled **SYMB** will change the emphasis from the graph to the equation in the right hand half of the screen. Pressing **PLOT** on the calculator, or the key labeled **GRPH** will change the emphasis from the equation back to the graph.

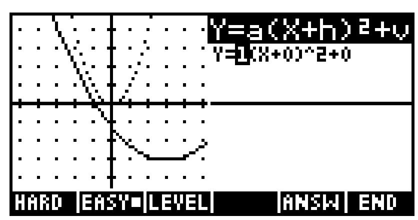
The alternative to graphical mode is symbolic mode. In **SYMB** mode the student can change the coefficients directly and see the changes reflected in the graph.



For example, on the screen shown below, the 'v' coefficient is high-lighted and can be changed using the ▲ and ▼ keys. The size of the changes to 'h' and 'v' is controlled by the **INC 1** key, and the same range of values is available for the values of 'a' as in **GRPH** mode. You can move from one coefficient to the next by using the ► and ◄ keys.

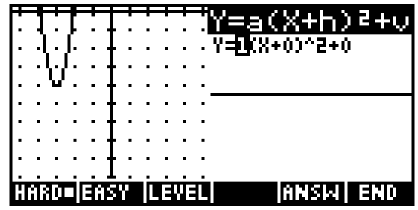


As with the **GRPH** mode, the equation in $y=ax^2+bx+c$ form, its roots and its discriminant are shown on the right half of the screen.

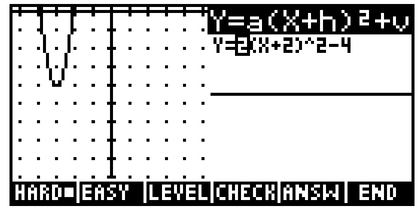


The + and - keys are disabled in **SYMB** mode, since their effects are controlled instead by the ▲ and ▼ keys once the highlight is on the 'a' coefficient. The (-) key controls the sign of 'a'.

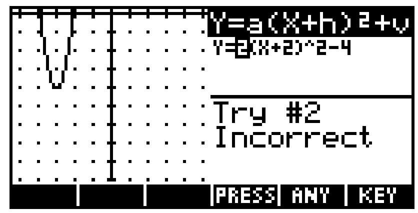
The final key is labeled **TEST**. This key will present the student with a series of graphs for which they must supply the equation. The type of graph is governed by the current setting of **LEVEL**. For example, if the current setting of **LEVEL** was at $Y = X^2 + v$ then the test graphs would also only use the *v* parameter instead of *a*, *h* and *v* simultaneously. The setting of **LEVEL** can be changed within the **TEST** screen using the key supplied.



There are two levels of 'questions' denoted by the keys **EASY■** and **HARD■** on the screen. An **EASY■** question will be in the main screen (-5 to 5 on each axis), whereas a **HARD■** one can be anywhere in the larger screen (see right). The setting of **LEVEL** also affects the difficulty of the question. The first is always **EASY■** but you can substitute another by pressing **EASY■** or **HARD■**.



In **TEST** mode you must use the arrow keys to change the parameters '*a*', '*h*' and '*v*' until they match the graph shown. The accuracy of your answer can be checked by pressing the key labeled **CHECK**, which appears as soon as you begin to change the values. The number of attempts is monitored and displayed.



An incorrect answer is shown as an 'animated' graph on the screen when you press **CHECK**, flashing repeatedly between the required graph and your incorrect guess. This has to be seen to be appreciated - a screen shot can't do it justice. If your guess and the required graph can't be shown on the same screen then this animation may not be possible. If you are unable to find the answer, pressing **ANSW** will display the correct parameters.

When you are successful, or when you give up, press either **EASY■** or **HARD■** for a new graph, or **END** to return to the main screen.

If you go to HP's website you can download a worksheet for use with your class. It takes the student through the process of deducing the effects of each of the coefficients on the shape of the graph, requiring them to record their answers in writing.

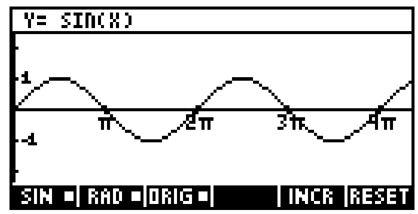
THE TRIG EXPLORER TEACHING APLET

Rather than being a multi-purpose aplet like most of the others covered so far, this is a teaching aplet specialized to the single use of exploring the graphs of trigonometric functions. As such it does not have the normal **SYMB**, **NUM** and **PLOT** views, but only a single multi-purpose view. The **SYMB** and **PLOT** keys do have meanings but not the normal ones.

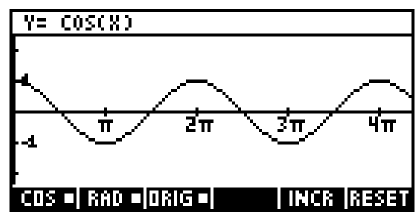
Objectives

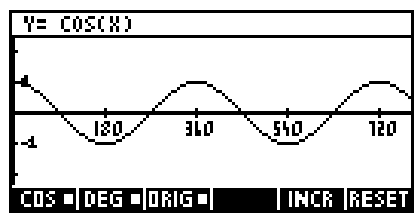
Using the Trig Explorer aplet, the student will investigate the behavior of the graph of $y = a\sin(bx + c) + d$ as the values of *a*, *b*, *c* and *d* change. This can be done both by manipulating the equation and seeing the change in the graph, *or* by manipulating the graph and seeing the change in the equation.

When the student presses **START**, they will see the main screen of the aplet, shown right. Like the Quadratic Explorer, this aplet has only one screen, rather than the usual **PLOT**, **SYMB** and **NUM** views. If the aplet has been used before then it may be necessary to press the **RESET** screen key to go back to the opening view shown.



The **SIN ■** (or **COS ■**) key can be used to toggle between the $y = \sin(x)$ curve and the $y = \cos(x)$ curve. The **RAD ■** (or **DEG ■**) key can be used to toggle between radian measure and degree measure. The markings on the horizontal axis adjust accordingly (see right and below right).





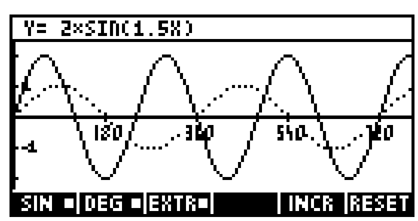
The Trig Explorer aplet can be used in two modes:

SYMB mode - in this mode the equation controls the graph (press the **SYMB** key).

and **PLOT** mode - in this mode the graph controls the equation (press the **PLOT** key).

Which mode is in operation is controlled by pressing the **SYMB** and the **PLOT** keys. The **NUM** key has no effect in this aplet and the aplet has no **SETUP** views. The ranges for the x and y axes are preset and cannot be changed.

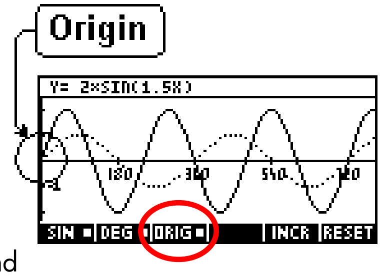
In both modes the original shape of the sine or cosine graph is left visible as a dotted line for comparison.



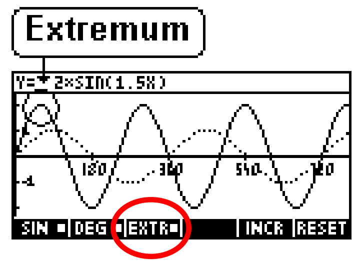
The operation of the two modes is summarized below.

The PLOT mode

The underlying concept in **PLOT** mode is that the graph controls the equation.



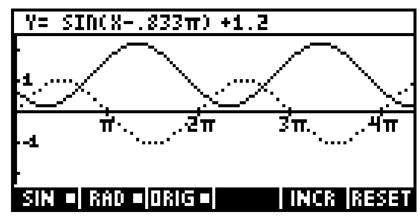
The user has control of the graph via two manipulation points (see above and below) and any changes to the graph are reflected in the equation at the top of the screen.



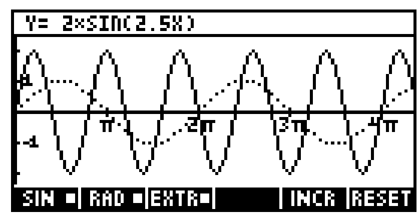
Looking at the screens on the right it will be seen that the third screen key toggles the point of control for the graph. It is only visible in **PLOT** mode.

When in **PLOT** mode the graph is changed using the arrow keys. The effects of the arrow keys vary according to the location of the control point.

- when control is set to **ORIG** the graph translates left and right (the *c* coefficient) or up & down (the *d* coefficient).



- when control is set to **EXTR** the graph dilates parallel to the y axis (the *a* coefficient) or parallel to the x axis (the *b* coefficient).

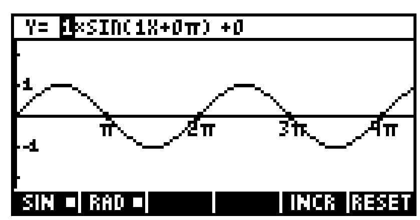


Unlike the **SYMB** mode where the *b* coefficient goes up or down in steps of 0.1, the increments in **PLOT** mode are not regular but are chosen to be 'nice' fractions of π. The slight difference in results is most easily seen by simply experimenting.

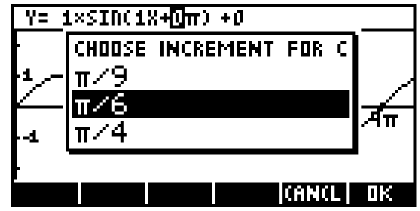
The SYMB mode

The underlying concept in **SYMB** mode is that the equation controls the graph. The user has control of the coefficients and any changes are reflected in the graph.

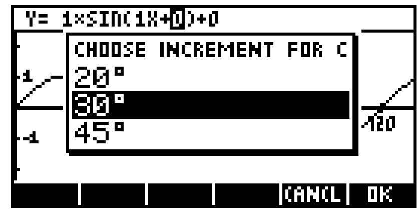
All four coefficients are shown at the top of the screen even when one or more is redundant.



The *c* coefficient is shown as a multiple of π in radian mode rather than as a decimal. The currently active coefficient is highlighted and can be changed using the up/down arrow keys in increments of 0.1 for the coefficients *a*, *b* and *d*.



The default increment for *c* is $\pi/6$ but this can be changed using the key labeled **INCR** to either $\pi/9$ or $\pi/4$.



When in **DEG** mode, the increments are 20°, 30° or 45° with 30° being the default.

The values available for the four coefficients are restricted to a given range and only certain values within that range. This is done to make the aplet more efficient and faster. The range of values is shown below:

Coefficient	Range
a	-3 to 3 in steps of 0.1
b	0.3 to 5 in steps of 0.1
c	-4π to 4π in steps of $\pi/6$ (or whatever is set by **INCR**)
d	-3 to 3 in steps of 0.1

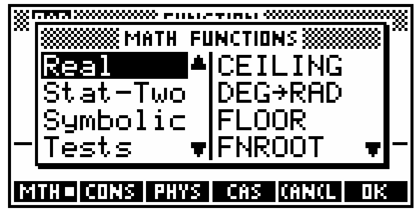
If you go to Hewlett Packard's website you can download a worksheet for use with your class. It takes the student through the process of deducing the effects of each of the coefficients on the shape of the graph, requiring them to record their answers in writing. The same worksheet can be found on the author's site at http://www.hphomeview.com

THE MATH MENUS

The **MATH** menu is accessed via the key below the **APLET** key. Any time that you are typing a value into any formula or setup screen you can insert mathematical functions via the **MATH** key.

In reality the **MATH** button gives you access not just to one menu but to four. The menus are:

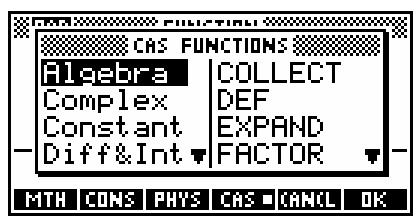
- The **MATH** menu - this is the default menu containing the majority of commonly used commands. The commands in this list are available on both the hp 39gs and hp 40gs.



- The **CONS** menu - this 'constants' menu is only really of interest to programmers. It contains the names of various environment variables, allowing easy pasting into programs. Generally it is easier to simply type them.

- The **PHYS** menu - this 'physical constants' menu contains the values of many of the physical constants required by students in physics and chemistry.

- The **CAS** menu - this menu contains those commands which are only available on the hp 40gs and are associated with the CAS. On the hp 39gs this screen key is blank and typing the command manually will result in an 'unknown user function' error.



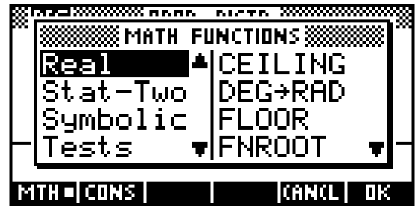
Since they are not relevant for both calculators, the CAS commands are covered in the section on the CAS for the hp 40gs.

The following pages will cover:

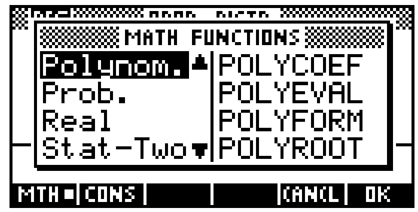
- how to use and access the commands
- the PHYS commands
- the MATH commands

Accessing the MATH menu commands

The mechanics of accessing the **MATH** menu is very simple. We will illustrate the process using the Polynomial function **POLYFORM**, which is an extremely useful one. Change into the **HOME** view and then press the **MATH** key.

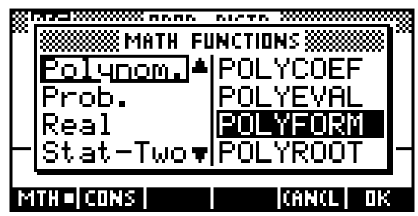


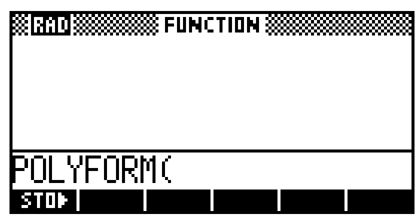
When you do you will see the screen on the right. The menu always first appears with the Real functions highlighted.



We could use the arrow keys to scroll down to the Polynomial functions but it is far faster to simply press the key labeled with the letter '**P**' (on the '**5**' key). It is not necessary to press the **ALPHA** key first. You will notice in the screen on the right that there are two groups of functions beginning with a '**P**' - being Polynomial and Probability. To reach Probability you would press the '**P**' key again or ▼.

Once you are in the correct group, press the right arrow key to move into the list of functions belonging to that group. Once again you have a choice of using the arrow keys or the key corresponding to the first letter of the function. Pressing the right arrow key again will move down the functions a 'page' at a time. In this case, since every single function in the Polynomial group begins with a '**P**', and there is only one 'page' of them, there is no difference between the methods. Move the highlight down to **POLYFORM** and then press the **ENTER** key.

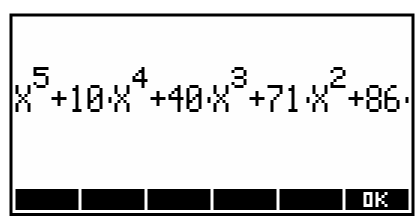




Your **HOME** view should now look like this. You will notice that the first bracket has already been inserted for you.

Complete the expression…

POLYFORM((X+2)^5-(3X-1)^2,X)
(using the x^y key to get **^**)

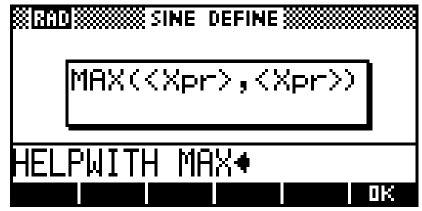


and then press the **ENTER** key. You will find that the expression $(x+2)^5-(3x-1)^2$ has been expanded on the following line to **X^5+10*X^4+40*X^3+71***…etc

There are two ways of seeing the complete result. You can move the highlight up to that line and **COPY** it. You could then move back and forth through the line using the arrow keys to see it in full.   A better method is to press the screen key labeled **SHOW**. This gives (after a pause) the result shown above. The portion off the edge of the screen can be seen by scrolling right using the arrow key. More details will be given of this **POLYFORM** function in the *Polynomial* group when we get to it.

On the pages which follow we will look at most of the functions in each group. Some of the functions are not likely to be used at school level and so will not be covered since this book is primarily aimed at teachers and students of high school, as are the hp 39gs & hp 40gs. If you need the higher level commands then consult the manual.

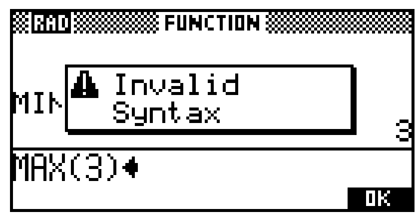
You can obtain 'help' for any function in the **HOME** view by using the **SYNTAX** key to obtain the word **HELPWITH** and then typing the function name. An example is shown right for the **MAX** command. Thus this is saying that you have to type in two expressions, which can be simply numbers but might be algebraic. As you can see, the custom is to enclose arguments in < > brackets and this is the custom followed in this text too during the later list of functions. When you use the function you don't actually include those brackets, just the argument. For example you would write **MAX(3.5,2A+5)** not **MAX(<3.5>,<2A+5>)**.



There is a limit to how much this **HELPWITH** statement will aid a normal user of the calculator, since these syntax statements are usually more suited to a programmer type than to a student. However they may be enough to trigger your memory.

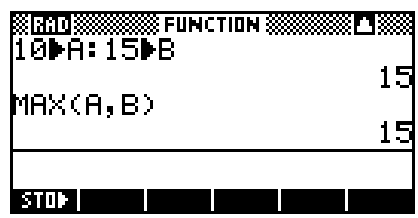
One piece of terminology that will be used in this section of the manual is 'argument'. The arguments of a function are the pieces of information it is expecting you to feed it before it will give you an answer. These might be numbers, variable names, lists, matrices or algebraic expressions.

The calculator will not guess what you mean. If you don't feed it the information it requires then it will simply give you an error message.



For example, the function **MAX** expects two arguments, both of them numbers or expressions. Feeding it only one (or more than two) will produce the result shown right.

These numbers can also be the contents of memories. Suppose you have stored 10 in memory **A** and 15 in **B**. Then **MAX(A,B)** will give 15.



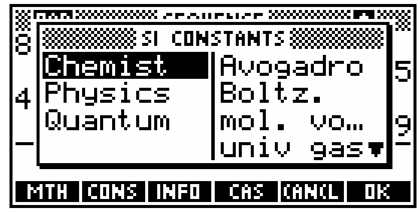
The commands available on the calculator are of two types. The type found in the **MATH** menu requires brackets to enclose its arguments and in this type of command the arguments are separated by commas.

Examples are: **MAX(15,X)** or **POLYFORM(2(X+3)^3,X)** or **ROUND(3.4465,3)**

The second type is more common in programming commands. It doesn't enclose its arguments in brackets and uses semi-colons to separate them. See the chapter on Programming for information on these commands. They are found in the menu that pops up when you press **SHIFT CMDS**.

Examples are: **DISPXY 3;5.4;1;"Hello"** or **BOX 1;1;-4;3**

The PHYS menu commands



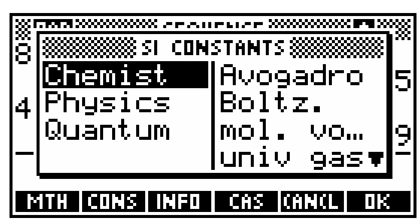
The **PHYS** menu is divided up into three sections by learning area. These sections are:

- Chemistry
- Physics
- Quantum Physics

The contents are simply the numerical values of various physical constants that are useful in calculations and formulae.

Chemistry	***Physics***	***Quantum Physics***
• Avogadro's number	• Stephan-Boltzmann constant	• Planck's constant
• Boltzmann's constant	• Speed of light (m/sec)	• Dirac's constant
• Molar volume	• Permittivity	• Charge on an electron
• Universal gas constant	• Permeability	• Mass of an electron
• Standard temperature	• Acceleration due to gravity	• q/m_e ratio
• Standard pressure	• Gravitational constant	• Mass of a proton
		• m_p/m_e ratio
		• fine structure constant
		• magnetic flux constant
		• Faraday constant
		• Rydberg constant
		• Bohr radius
		• Bohr magneton constant
		• Nuclear magneton const.
		• Photon wavelength
		• Photon frequency
		• Compton wavelength

At the bottom of the screen is an **INFO** button which, when pressed, gives more information on the particular constant chosen.



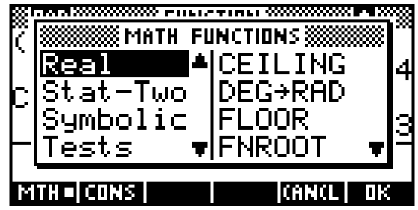
The MATH menu commands

The **MATH** menu is divided up into sections by mathematical topics. These topics are:

Real	- rounding, roots, some conversions and percentage functions.
Stat-Two	- bivariate statistical functions.
Symbolic	- functions for manipulating equations and symbols.
Tests	- used in programming more than normal work.
Trig	- contains the trig functions not found on the keyboard. E.g. sec, cosec.
Calculus	- integration and differentiation
Complex	- functions to manipulate complex numbers.
Constants	- e, i and various others. Of more use in programming.
Convert	- contains a list of functions allowing conversion of units.
Hyperb.	- the hyperbolic trig functions.
List	- allows manipulation & creation of lists of numbers, including stats data.
Loop	- iterative functions.
Matrix	- a rich collection of functions to manipulate matrices.
Polynom.	- another rich collection, this time to manipulate polynomials.
Prob.	- functions used in probability calculations.

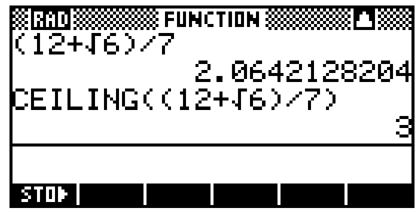
Some of these functions have little application at school level and will not be covered here. Others will be covered to varying depths. Anyone needing those not covered will find them in the manual that comes with the calculator.

The ‘Real’ group of functions



CEILING(<num>)

This is a ‘rounding’ function but different in that it always rounds up to the integer above. Mainly of interest to programmers.

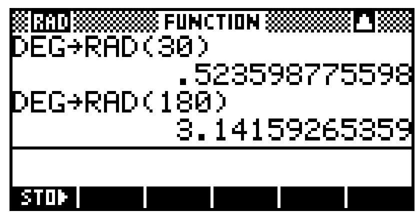


Eg. **CEILING(3.2)** = 4
CEILING(32.99) = 33
CEILING((12+√6)/7) = **CEILING(2.0642...)** = 3

Note: **CEILING(-2.56)** = -2 not -3. The **CEILING** function rounds up to the next integer *above*, which is -2.

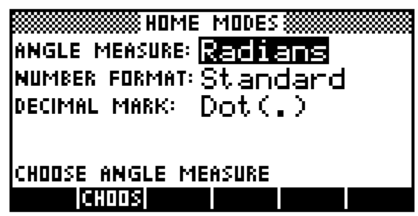
See also: **ROUND**, **TRUNCATE**, **FLOOR**, **INT**

DEG→RAD(<deg>)



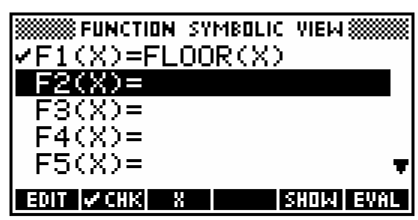
This function converts degrees to radians.

Eg. **DEG→RAD(30)** = 0.5235...
DEG→RAD(180) = 3.1415926...



See also: **RAD→DEG**, **HMS→**, **→HMS**

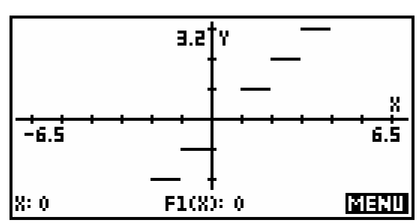
FLOOR(<num>)



This function is the same as the **CEILING** function, except that it always rounds down.

Eg. **FLOOR(3.75)** = 3
FLOOR(45.01) = 45

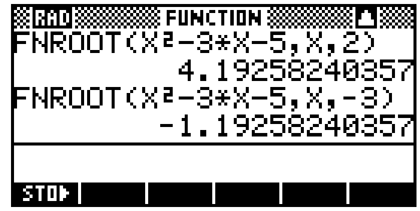
Note: **FLOOR(-2.56)** = -3 not -2.



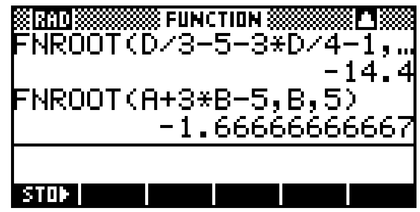
The **FLOOR** function is the same as the mathematical function ‘greatest integer’ which is studied in many mathematical courses.

If you want to graph the greatest integer function then you will need to use the **PLOT SETUP** view to turn off **CONNECT** first, since the graph is supposed to be a discontinuous step function. The result with **CONNECT** turned on is not good.

FNROOT(<expression>,<variable>,<guess>)



This function is like a mini version of the Solve aplet. If you feed it an algebraic expression and an initial guess it will start from your guess and find the value which makes the expression zero. Don't bother. It's a lot easier to use the Solve aplet. This is a tool for programmers so that they can access the Solve abilities within programs.



You need to tell it what variable is the active one in the expression, in addition to providing it with an initial guess. If there is only one answer then any guess will do, but if more than one solution is possible then more care needs to be taken with your guess to ensure that it is in the neighborhood of your desired solution.

Eg. (a) Solve $x^2 - 3x - 5 = 0$

Use: **FNROOT(X²-3X-5,X,2)**

(b) Solve $\frac{d}{3} - 5 = \frac{3d}{4} + 1$

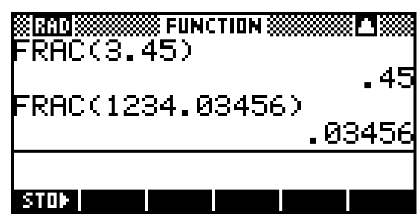
Use: **FNROOT(D/3-5-3D/4-1,D,0)**

As with Solve, if your expression involves more than one variable then whatever values are currently in memory are used for the other variables.

See also: **QUAD**, **POLYROOT**

FRAC(<num>)

This function gives the decimal part of any number, discarding the integer part.

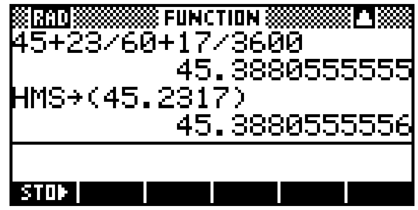


Eg. **FRAC(3.45)** = 0.45
FRAC(1234.03456) = 0.03456

See also: **INT**,**FLOOR**,**CEILING**,**ROUND**,**TRUNCATE**,**FRAC**

HMS→ (<dd.mmss>)

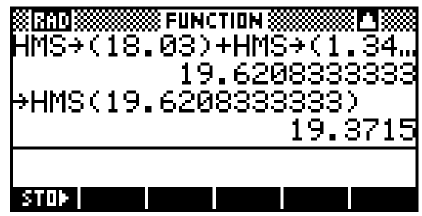
This function works with time and angles. It converts degrees, minutes and seconds to degrees, and also hours, minutes and seconds to decimal time.



The calculator can convert a value such as $45^\circ 23' 17''$ if you put it into the form 45.2317 and then use the **HMS→** function.

E.g. sin($45^\circ 23' 17''$) would be **SIN(HMS→(45.2317))**
cos($5^\circ 3' 7''$) would be **COS(HMS→(5.0307))**

This function, together with **→HMS**, can also be used to deal with time.



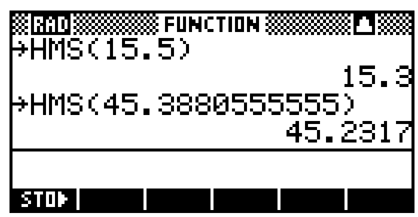
E.g. What time will it be 1 hr 34 min. and 15 sec. after 3 min. past 6 pm?

Type: **HMS→(18.03)+HMS→(1.3415)**

Ans: 37 min. 15 sec. past 7 pm. (see right)

See also: **→HMS**, **RAD→DEG**, **DEG→RAD**

→HMS(<num>)

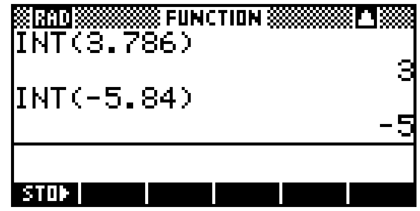


This function works in the same way as the **HMS→** function but in the opposite direction. It converts decimal degrees (or time) to degrees (or hours), minutes and seconds. The format of the returned answer is DD.MMSS (or HH.MMSS if dealing with time)

Eg. $15{\cdot}5^\circ$ would become $15{\cdot}3$, meaning $15^\circ 30'$.
45.38805555 → 45.2317 ($45^\circ 23' 17''$)
3.75 hours → 3.45 (3 hours 45 min.)

See also: **HMS→**, **RAD→DEG**, **DEG→RAD**

INT(<num>)

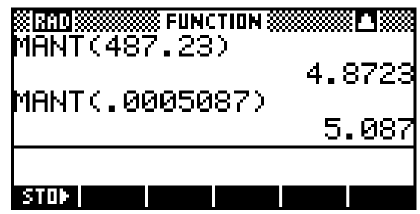


This function is related to the **FLOOR** and **CEILING** functions. Unlike those two, which consistently move down or up respectively, the **INT** function simply drops the fractional part of the number.

Eg. **INT(3.786)** = 3
INT(-5.84) = -5

See also: **FLOOR**, **CEILING**, **ROUND**, **TRUNCATE**, **FRAC**

MANT(<num>)



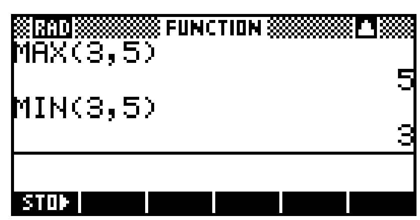
This function returns the mantissa (numerical part) of the number you feed it when transformed into scientific notation. It would be used with the **XPON** function, which returns the power part of the number in scientific notation.

Eg. Change $487 \cdot 23$ into scientific notation to get $4 \cdot 8723 \times 10^2$.
MANT(487.23) = 4.8723
XPON(487.23) = 2

Change $0 \cdot 0005087$ into scientific notation to get $5 \cdot 087 \times 10^{-4}$.
MANT(0.0005087) = 5.087
XPON(0.0005087) = -4

See also: **XPON**

MAX(num1,num2)

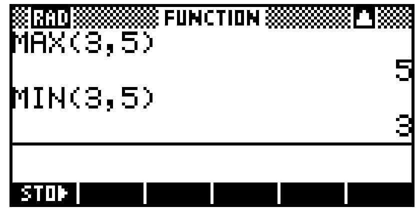


This function returns the larger of two values entered. This is not needed in your normal calculations, since you could just look at the numbers, but a programmer will be writing a program which deals with numbers not known in advance.

Eg. **MAX(3,5)** = 5

See also: **MIN**

MIN(num1,num2)

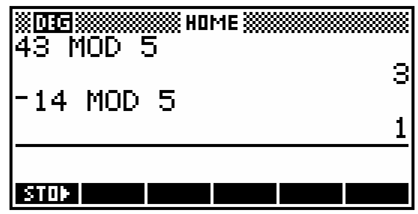


As with **MAX**, this function is used mainly by programmers. It returns the smaller of the two numbers entered.

Eg. **MIN(3,5)** = 3

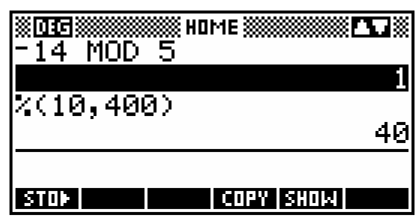
See also: **MAX**

<num> MOD <divisor>



For those not familiar with modulo arithmetic, it will suffice to say that this function gives you the remainder when one number is divided by another. It is considered to be an mathematical operator in the same way that a plus, minus, times or divide sign is. Because of this it does not need its arguments placed in brackets as most of the other functions in the **MATH** menu do.

Eg. **43 + 5** = 48
43 MOD 5 = 3
35 MOD 7 = 0
-14 MOD 5 = 1



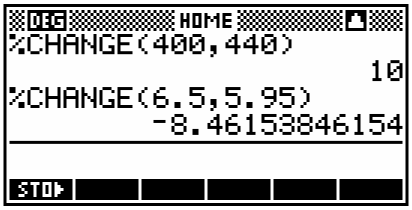
% function(<num1>,<num2>)

To find x% of y, use the function **%(X, Y).**

Eg. 10 % of $400 = $40 is shown right.

See also: **%**, **%TOTAL**, **%CHANGE**

%CHANGE(<num1>,<num2>)



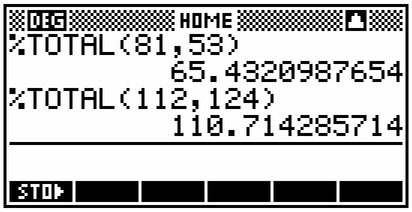
This function calculates the percentage change moving from **X** to **Y** using the formula **100(Y-X)/X**. It can be used to calculate (for example) percentage profit and loss.

Eg. I buy a fridge for $400 and sell it for $440.
What is my profit as a percentage?
Use: **%CHANGE(400,440)**

I sell a toy for $5.95 that normally sells for $6.50
What is the discount as a percentage of the usual price?
Use: **%CHANGE(6.50,5.95)**

See also: %, **%TOTAL**

%TOTAL(<num1>,<num2>)



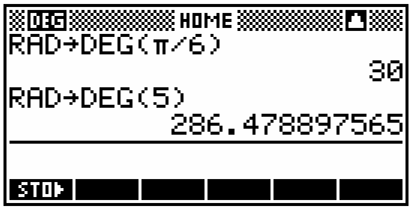
To find out what percentage **X** is of **Y**, use the function **%TOTAL(Y,X)**. Note the reversed order.

Eg. What percentage is a score of 53 out of 81 on a test?
Use: **%TOTAL(81,53)**

What percentage is 124 of 112?
Use: **%TOTAL(112,124)**

See also: %, **%CHANGE**

RAD→DEG(<radian_measure>)

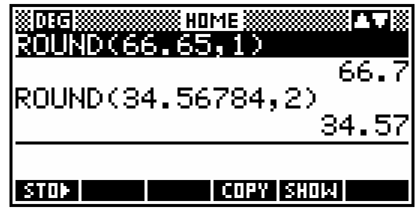


This function converts radians to degrees.

Eg. **RAD→DEG(π/6)** = 30°
RAD→DEG(5) = 286.48°

See also: **DEG→RAD**, **HMS→**, **→HMS**

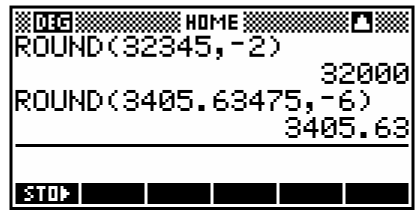
ROUND(<num>,<dec.pts>)



This function rounds off a supplied number to the specified number of decimal places (d.p.).

Eg. Round 66.65 to 1 d.p.
Use: **ROUND(66.65,1)** = 66.7

Round 34.56784 to 2 d.p.
Use: **ROUND(34.56784,2)** = 34.57



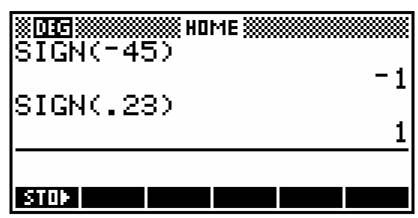
This function is also capable of rounding off to a specified number of significant figures (s.f.). To do this, simply put a negative sign on the second argument.

Round 32345 to the nearest thousand.
Use: **ROUND(32345,-2)** = 32000

Round 3405.63475 to 6 s.f.
Use: **ROUND(3405.63475,-6)** = 3405.63

See also: **INT**, **FLOOR**, **CEILING**, **TRUNCATE**, **FRAC**

SIGN(<num>)



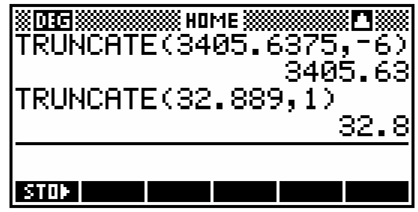
Another function designed more for programmers, returning a value of +1, 0 or -1 depending on whether the number supplied is positive, zero or negative.

Eg. **SIGN(-45)** = -1

See also: **XPON**, **MANT**

Note: See also **SIGN** in the Complex group of functions (page 187).

TRUNCATE(<num>)



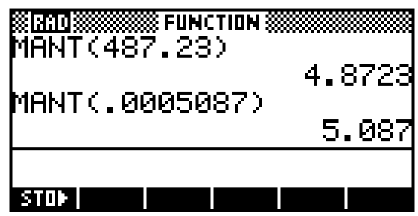
This function operates similarly to the **ROUND** function, but simply drops the extra digits instead of rounding up or down. It is somewhat similar in effect to the **FLOOR** function but the **TRUNCATE** function will work to any number of decimal places or significant figures instead of always dropping to the nearest lower integer value.

Eg. **TRUNCATE(3405.6375,-6)** = 3405.63

TRUNCATE(32.889,1) = 32.8

See also: **INT**, **FLOOR**, **CEILING**, **ROUND**,**FRAC**

XPON(<num>)



This function returns the exponent (indicial part) when transformed into scientific notation of the number you feed it. It would be used with the **XPON** function, which returns the power part of the number when in scientific notation.

Eg. Change $487 \cdot 23$ into scientific notation to get $4 \cdot 8723 \times 10^{2}$.

MANT(487.23) = 4.8723
XPON(487.23) = 2

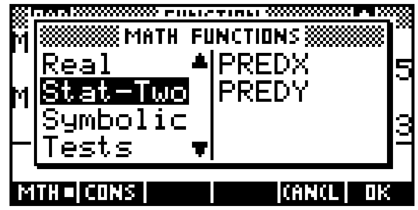
Change $0 \cdot 0005087$ into scientific notation to get $5 \cdot 087 \times 10^{-4}$.

MANT(0.0005087) = 5.087
XPON(0.0005087) = -4

This function could be of use to you if you are just learning scientific notation, but is of more use to people writing programs. A normal user would just look at the number and see the answer, but a programmer would not know in advance what number was going to be used and so might use **MANT** and **XPON** to find the size of a number that the user has just entered.

See also: **MANT**

The 'Stat-Two' group of functions



PREDY(<x-value>)

This function predicts the y value for a pair of columns set up as bivariate data in the Statistics aplet. This is discussed in more detail in the section covering the Statistics aplet, but a brief summary will be given here.

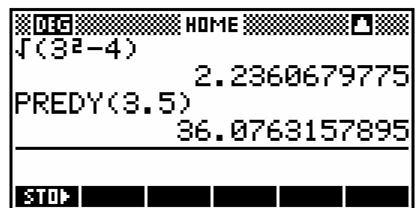
It assumes that:

(i) the bivariate data is entered into a pair of columns (eg. **C1** and **C2**, with **C1** containing the independent data and **C2** the dependent data),

and (ii) that these two columns have been specified in the **SYMB** view to be paired bivariate data,

and (iii) that this data set has been graphed in the **PLOT** view and that the **FIT** screen key has been used to plot the line of best fit for the pair of columns,

If these conditions are satisfied then the function **PREDY(3.5)** will produce a predicted y (dependent) value for the x (indep.) value of $3 \cdot 5$.



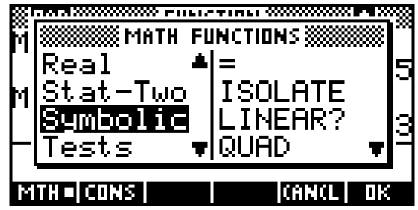
> **Calculator Tip**
> The line of best fit used in the function **PREDY** is whichever one was last plotted. *It is up to you* to ensure that this is in fact the one you want used! It is far easier and more reliable to use the facilities provided in the **PLOT** view (see page 130) to calculate **PREDY** values.

PREDX(<y-value>)

The **PREDX** function simply reverses the line of best fit. For example, the equation $\hat{Y} = 0 \cdot 8199X + 1 \cdot 1662$ earlier would use $\hat{X} = \dfrac{(Y - 1.1662)}{0.8199}$ to predict the X values.

Note: See page 136 for very important information on this function.

The 'Symbolic' group of functions

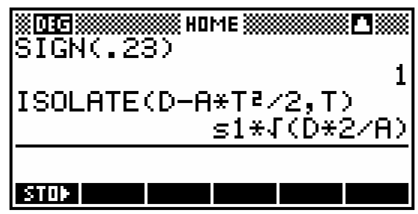


The = 'function'

Although this is listed in the **MATH** menu as if it were a function, it is not really. Except in programming, the = sign is simply used in exactly the way that you would expect it to be, mainly in the Solve aplet. It's easier to obtain the = sign directly from the keyboard. The reason that it is found in the **MATH** menu is that the original calculator, the hp 38g, had less keys on the keyboard and had no room for it. It was added in later models.

ISOLATE(<expression>,<var-name>)

This function will rearrange a formula so that its subject is another variable. To do this, the formula must be rewritten so that it is an expression which equals zero. The **ISOLATE** function then rearranges the formula in terms of that variable.



Eg. 1 Rewrite the formula $d = \frac{1}{2}at^2$ in 't'.

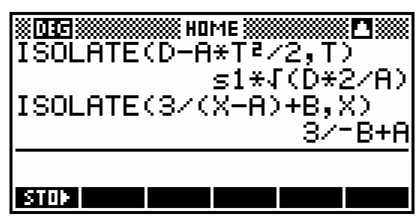
Firstly, rewrite as: $d - \frac{1}{2}at^2 = 0$

Use: **ISOLATE(D-AT2/2,T)**

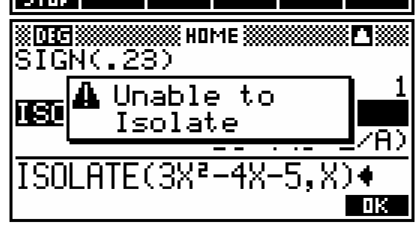
The result needs interpretation as the answer should be $T = \pm\sqrt{\frac{2D}{A}}$. Firstly, the 'T=' is missing and secondly, the symbol '**S1**' stands for the $\pm$ sign.

Eg. 2 Make **X** the subject of the formula $B = \frac{-3}{(X-A)}$.

Using: **ISOLATE(3/(X-A)+B,X)**
gives: **3/-B+A**
equivalent to: $X = \frac{-3}{B} + A$



The **ISOLATE** function is useful within its limitations, but it will only deal with a limited set of formulae (see right). The CAS on the hp 40gs is far more flexible.

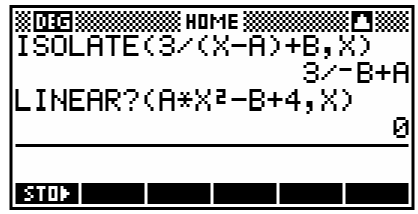


See also: The Solve aplet, **FNROOT**, **QUAD**, The CAS chapter

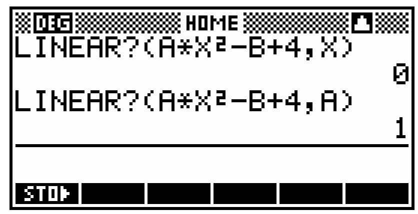
LINEAR?(<expression>,<var.name>)

This is another of those functions which is probably aimed more at the programmer than at the normal user. It is designed to test whether a supplied expression is linear or non-linear in the variable you specify, returning zero for non-linear and 1 for linear.

Eg. Suppose we use the expression $AX^2 - B + 4$



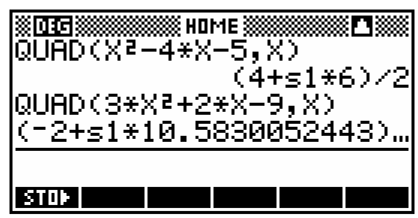
If X is the variable and A and B are both constants (say **A=4, B=5**) then the expression $AX^2 - B + 4$ would become $4X^2 - 5 + 4$ which would be non-linear. Thus **LINEAR?** returns zero (right).



On the other hand, if X were one of the constants (say **X=6**) and A were the variable, then the expression $AX^2 - B + 4$ would become $A \times 6^2 - 5 + 4$ or $36A - 1$, which is linear. Thus **LINEAR?** would return a value of **1** as shown right.

The main use for this is going to be when a programmer does not know in advance what function the user is going to type in.

QUAD(<expression>,<var.name>)



This function uses the quadratic formula $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ to give both roots of a quadratic, using the '**S1**' formal variable to represent the $\pm$ symbol. The quadratic is entered as an expression, and you must indicate which variable is being solved for, since you could have an equation such as $Px^2 + Qx - 5 = 0$ where **P** and **Q** were memory values, and you would need to specify to solve for **X** in order to tell the calculator that the active variable was **X** and not **P** or **Q**.

Eg. Solve $x^2 - 4x - 5 = 0$
Use **QUAD(X2-4X-5,X)**
Answer: **(4+S1*6)/2**

It is now up to you to interpret this algebraically as:

$$x = \frac{4 \pm 6}{2}$$
$$= \frac{4+6}{2} \text{ or } \frac{4-6}{2}$$
$$= 5 \text{ or } -1$$

If you are simply after the roots of the quadratic then it is far better to use the **POLYROOT** function (page 298) or to graph the function and use the **FCN** tools.

If you would like a solution such as $\frac{3+\sqrt{5}}{2}$ rather than 2.6180 then you would have to **COPY** the result, edit the line to remove all but the decimal root and square it to find the original discriminant.

If you are fortunate enough to have an hp 40gs rather than an hp 39gs then you can do all this far more easily in the CAS. See page 309 for details on finding roots of real and complex polynomials using the CAS on the hp 40gs.

See also: FNROOT, LINEAR?

QUOTE(<var_name>)

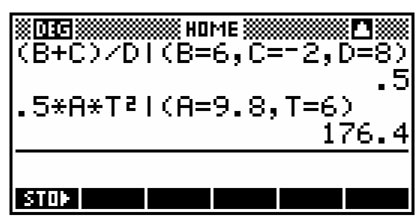
Again, this is a function intended for use mainly by programmers. Programmers sometimes want to store a function such as **X^2 - 4** into one of **F1(X)**…**F9(X)** using **STO▶**. It turns out that if you use **F1(X^2 - 4**) then it won't be entered symbolically. Instead, the contents of memory **X** (a number) is substituted and entered and then the expression is evaluated to give a numeric result. The **QUOTE** function fixes this. For example, **QUOTE(X)^2 - 4▶F1(X)** will ensure a symbolic result.

An easier method of storing a function into an aplet in a program is to enclose it in single quotes. For example **'(X)^2 - 4'▶F1(X)** would serve the same purpose as **QUOTE(X)^2 - 4▶F1(X).** On the other hand, entering **F1('X')** will not work but **F1(QUOTE(X))** will. No-one ever promised consistency! See Example 1 on page 262 in the chapter "Programming on the hp 39gs & hp 40gs" for an example of use in writing code.

The | function written as: <expression> | (var1=value,var2=value,…)

This is called the 'where' function. The reason for this is that it is used to evaluate formulas, of the type when one would say "Evaluate ….., where a = 5, b = 4 etc". The formula must be in the form of an expression rather than an equation. You should enter the expression first, then the 'where' symbol and then the values of all the variables in the expression. Any not supplied will be evaluated using the value currently stored in that memory. This is again a function which is of more use to programmers since this is definitely handled far more flexibly in the Solve aplet.

Eg. 1 Evaluate $a = \frac{b+c}{d}$ where b = 6, c = -2 and d = 8



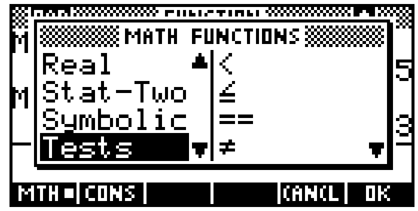
Use: **(B+C)/D|(B=6,C=-2,D=8)**
Answer: 0.5

Eg. 2 Using the formula $d = \frac{1}{2}at^2$, find the distance d which an object would fall under Earth's gravity of 9.8 m/s^2 in a time of 6 seconds.

Use: **.5AT2|(A=9.8,T=6)**
Answer: 176.4 meters

See also: ISOLATE

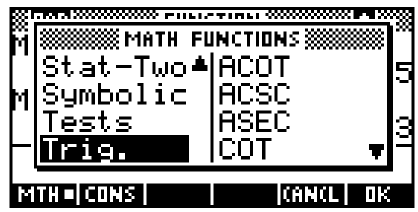
The ‘Tests’ group of functions



These are all functions which are of interest only to programmers, and consequently we will not cover them here. For anyone who has done any programming their use is obvious. They can also be found, far more conveniently, in the **CHARS** view.

A fairly thorough introduction to programming on the calculator is given in an later chapter (see page 255). Those wanting more detail than is given there must consult the manual. One way to learn more about programming is to download aplets from the internet and dismantle them to see how the code produces the results.

The ‘Trigonometric’ & ‘Hyperbolic’ groups of functions



These two groups of functions cover the Trigonometry functions, plus others, which are less commonly used and which have consequently not been given their own keys on the face of the calculator. Use them in the same way as the normal **SIN, COS** and **TAN** functions.

COT, SEC etc

In the Trig. group of functions you will find:

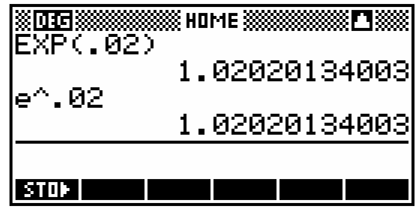
Function	Inverse function
COT (cotangent)	**ACOT** (arc-cotangent)
CSC (cosec/cosecant)	**ACSC** (arc-cosec)
SEC (secant)	**ASEC** (arc-secant)

In the Hyperb. group of functions you will find (amongst other things):

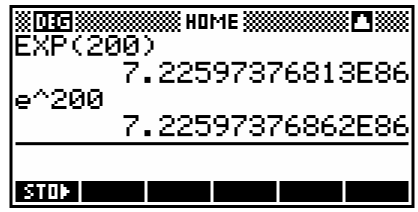
Function	Inverse function
SINH (hyperbolic sine or “shine”)	**ASINH** (arc-hyp.sine)
COSH (hyperbolic cos or “cosh”)	**ACOSH** (arc-hyp.cos)
TANH (hyperbolic tan)	**ATANH** (arc-hyp.tan)

Some further functions are available in the Hyperbolic group of functions. They are duplicates of functions available on the face of the calculator but give more accurate answers. They would primarily be of use to those people, such as architects and engineers, for whom high accuracy is paramount. These are:

EXP(<num>)

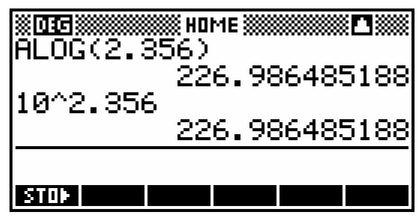


This function gives a more accurate answer than the key labeled **e^** which appears above the **LN** key on the calculator. As you can see on the right, the difference is normally not detectable even to 12 significant figures.



The difference is only apparent for some values and even then is hardly earth-shattering.

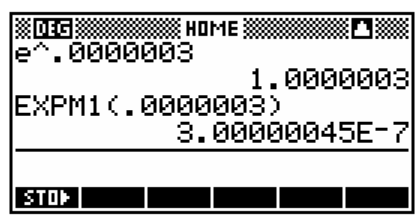
ALOG(<num>)



This function provides the same result as the key labeled **10^** on the keyboard above the **LOG** key. It is another function giving greater accuracy than the one it 'replaces'. This greater accuracy would probably never be required in a school setting.

EXPM1(<num>)

This function is designed to be more accurate when anti-logging very small values close to zero. It gives the value not of e^x but of $e^x - 1$ (**EXPM1** = *exp minus 1).* You may wonder how this is an advantage, since you must then add 1 to obtain the correct answer, but a look at the screen opposite will show you.



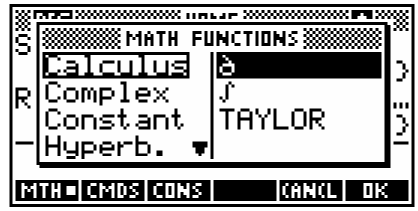
As you can see, the normal keyboard function **e^** gives an answer to $e^{0.0000003}$ of 1.0000003. This gives the impression that it is an exact value (since it doesn't show a full 12 significant digits). The true answer is 1·000000300000045…. but the final digits have been lost in the rounding off to 12 sig. figures.

By giving an answer of $e^x - 1$, the leading 1 is lost, freeing the calculator to show more accuracy by dropping the leading zeros. This is not normally be needed in the classroom.

LNP1(<num>)

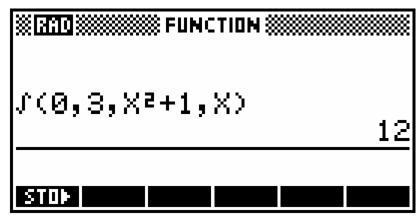
As in the previous function, this is supplied to supplement the **LN** function and gives a more accurate value when x is near zero. Again, this is not something which would normally be of concern at school level.

The 'Calculus' group of functions



This group consists of three functions, the integrate, or ∫ function, the differentiate, or ∂ function and the **TAYLOR** function. The first two are discussed in detail in the chapter dealing with the Function aplet (see pages 59 to 75) and so a brief outline only is given below.

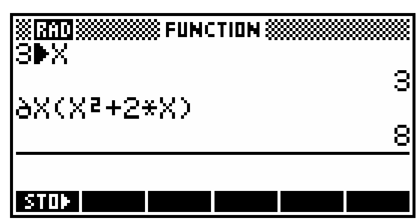
∫ (<num>,<num>,<expression>,<var_name>)



This function will return the definite integral of the expression when integrated with respect to the variable specified. Any other variables in the expression will be regarded as constants with values taken from the current memory values.

Symbolic integration can be done in two ways. Firstly by replacing one of the limits of integration with a symbolic variable **S1** (**S1…S5**). Secondly, and more conveniently, by doing it in the Function aplet (see pages 59 to 75).

∂<var_name>(<expression>)

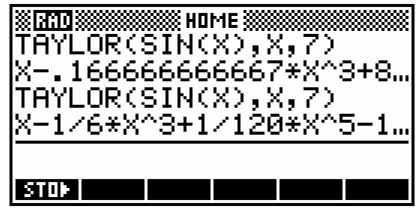


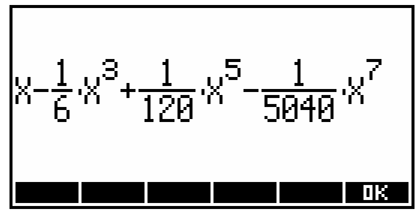
This function will differentiate the expression with respect to the variable specified. This can be done in two ways. When done in the **HOME** view the result is numeric because the derivative is evaluated for the current value of the variable in memory. For example, if **X** currently has the value of 3 then the result is as shown right.

When done in the Function aplet, or using a symbolic variable (**S1…S5**), the result is the algebraic derivative (see pages 59 to 75).

TAYLOR(<expression>,<var_name>,<num>)

Briefly, a Taylor polynomial allows you to approximate a complicated function via a simpler polynomial function. The <expression> supplied is approximated with respect to <var_name> by terms of a polynomial up to <num> power.



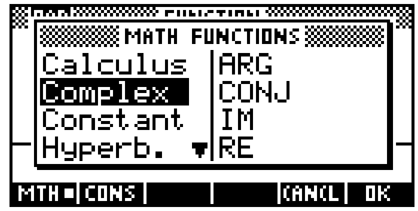


The screen shot on the right shows the calculator deriving the Taylor polynomial for sin(x) up to the 7th power. The **SIN(X)** function can be approximated by taking terms from the polynomial:

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + ...$$
$$= x - \frac{1}{6}x^3 + \frac{1}{120}x^5 - \frac{1}{5040}x^7 + ...$$

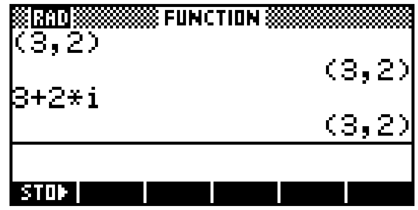
In this example, the result is shown twice. The first is calculated with **MODES** set to Standard, the second with **MODES** set to **Fraction 4**. The second screen shot shows the fractional polynomial in more detail after highlighting it and pressing **SHOW**.

The 'Complex' group of functions



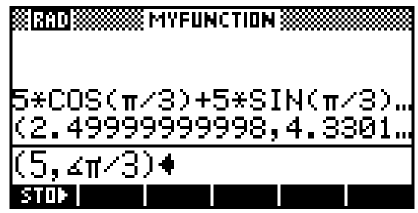
Complex numbers on the hp 39gs & hp 40gs can be entered in either of two ways. Firstly, in the same way as they are commonly written in mathematical workings: *a + bi*. Secondly, as an ordered pair: *(a,b).*

For example, $3+2i$ could be entered into the calculator exactly as it is written, with the '*i*' obtained using **SHIFT ALPHA** to get a lowercase *i*. Alternatively you can enter it directly as an ordered pair.



As soon as you press **ENTER**, the calculator immediately converts the *a + bi* form into an ordered pair. The History retains the original in case you need to **COPY** it later for re-use.

The exception is when you enter a complex number in polar form using the ∡ (angle) sign on the keyboard (above the [–] button). When you do this the calculator converts into the two other common forms of rcis format and (a,b) format as shown right. In the example right, the lowest line shows how the number was been entered. The first two lines show how the result is stored in the History as rcis format and as *(a,b)* format.

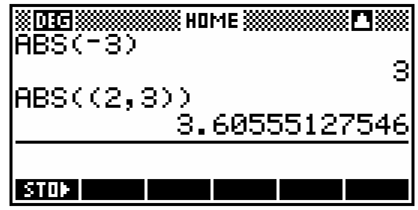


Complex numbers can be used with all trigonometric and hyperbolic function, as well as with matrices, lists and some real-number and keyboard functions.

Just as real numbers can be stored into the alphabetic memories A to Z, there are 10 special memories **Z1,Z2..Z9,Z0** which are provided to store complex numbers.

In addition to the trig functions, there are other functions that take complex arguments.

ABS(<real>) or ABS(<complex>)

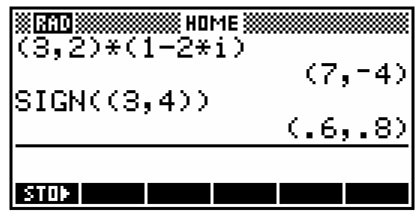


The absolute function, which is found on the keyboard above the left bracket key, returns the absolute value of a real number.

Eg. **ABS(-3)** returns a value of 3.

When you use the absolute function on a complex number *a + bi* it returns the magnitude of the complex number as $\sqrt{a^2+b^2}$. Note the requirement for double brackets in this case. The outer pair are those of the function **ABS(...)**. The inner pair are those of the number in *(a,b)* format.

SIGN(<real>) or SIGN(<complex>)

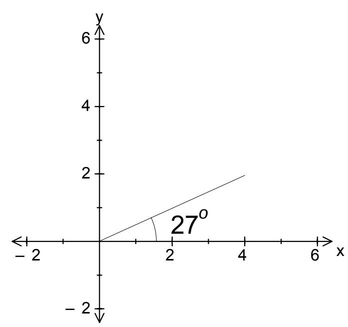


This function is found in the Real group not the Complex group but is very useful with complex numbers and so is also covered here. If given a vector/complex number *(a,b)*, **SIGN** will return another vector/complex number which is a unit vector in the direction of *(a, b)*. Note again the requirement for doubled brackets.

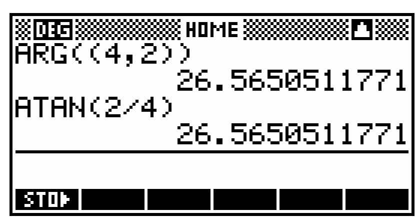
i.e. **SIGN((A,B))** returns $\left(\frac{a}{\sqrt{a^2+b^2}},\frac{b}{\sqrt{a^2+b^2}}\right)$.

This is very useful, not just in complex numbers, but also in vector problems.

See also: **SIGN** (in the Real group),**IM**,**RE**,**ARG**,**CONJ**

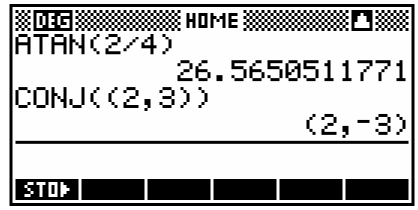


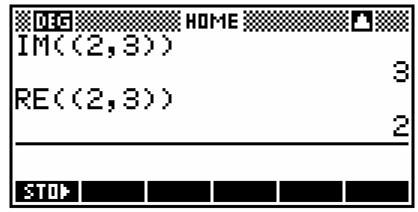
ARG(<complex>) or ARG(<vector>)



This function, also found on the keyboard, returns the size of the angle defined by regarding the complex number as a vector. For example **ARG**(*4+2i*) would be $26{\cdot}565^{\circ}$. The same information can, of course, be obtained using trig. The result is dependent on the current angle setting in **MODES**. Again, note the requirement for doubled brackets. Using **ARG(a+bi)** instead avoids this requirement.

CONJ(<complex>)





This function returns the complex conjugate.

Eg. If $z = 2 + 3i$, then find the complex conjugate $\overline{z}$.

Answer (see right): $\overline{z} = 2 - 3i$

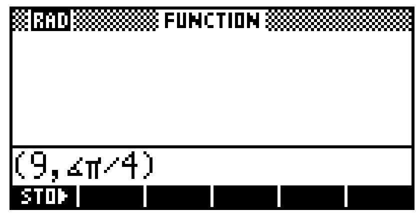
See also: **IM**,**ARG**,**RE**

IM(<complex>)

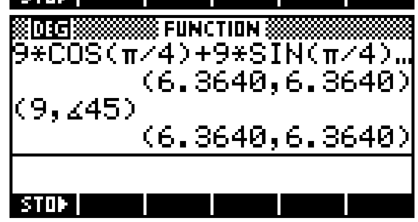
and RE(complex)

These functions return the imaginary and real parts of the complex number supplied.

See also: **CONJ**,**ARG**,**RE**

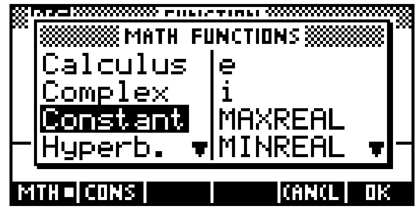


Note: As mentioned earlier, a very useful function (∡) can be found on the keyboard as the **SHIFT** function for [–]. If you enter a complex number in (r, θ) form as shown right, then the calculator will display it in the form $r\cos(\theta) + r\sin(\theta)$ from, and as an *(a,b)* ordered pair.

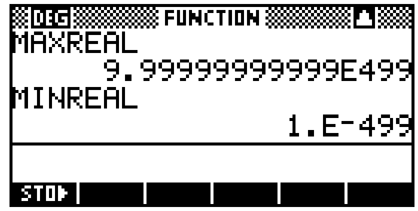


Note the $r\cos(\theta) + r\sin(\theta)$ form is used only when in radian mode. If in degree mode then it returns only the (a,b) form as shown right.

The 'Constant' group of functions



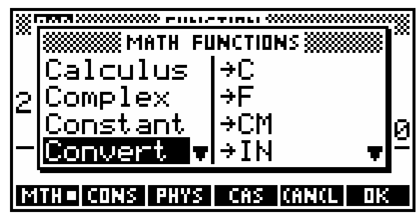
These 'functions' consist of a set of commonly occurring constants.



Two of them, **MAXREAL** and **MINREAL** are mainly of use to programmers except for an important influence on the evaluation of limits (see page 80). They consist, respectively, of the largest and smallest numbers with which the calculator is capable of dealing, and are there for use by programmers as a check to ensure that calculations within a program have not overflowed the capacity of the calculator.

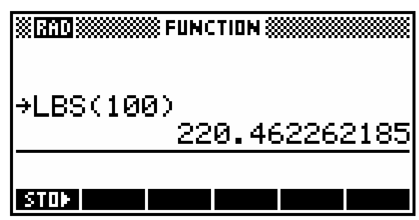
The other three, π, ***i***, and ***e***, are far more easily obtained via the keyboard.

The first, π, is available via a key on the face of the calculator above the 3 key. The other two , ***i*** & ***e***, are easily obtained as lowercase letters via the **ALPHA** key, pressing **SHIFT** first to get lowercase.



The 'Convert' group of functions

This is a set of functions which allow conversion of metric units to imperial and vice versa. The list is not very extensive but does cover the more common ones. The syntax is the same for all.

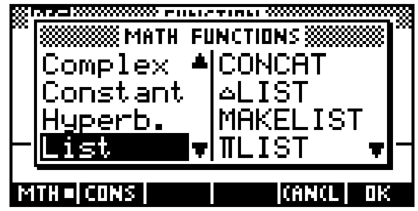


For example, the →**C** function converts to Celsius from the equivalent Fahrenheit. To convert 212 degrees Fahrenheit to Celsius just type →**C(212)**. To convert 100 kg to pounds, use →**LBS(100)** as shown in the screen shot.

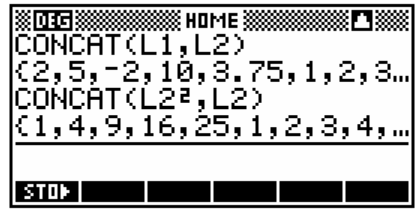
The available functions are:

→**C**	- converts Fahrenheit to Celsius	→**F**	- converts Celsius to Fahrenheit
→**CM**	- converts inches to centimeters	→**IN**	- converts centimeters to inches
→**L**	- converts litres to US gallons	→**GAL**	- converts US gallons to litres
→**KG**	- converts pounds to kilograms	→**LBS**	- converts kilograms to pounds
→**KM**	- converts miles to kilometers	→**MILE**	- converts kilometers to miles
→**DEG**	- converts radians into degrees	→**RAD**	- converts degrees into radians

The 'List' group of functions



CONCAT(<list1>, <list2>)



This function concatenates two lists - appending one on to the end of the other in the order that you specify. Lists must be enclosed in curly brackets unless list variables are used.

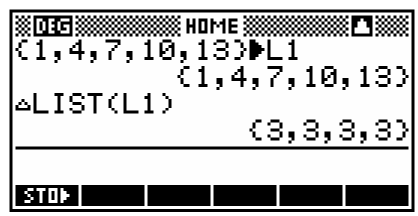
Eg. L1={2,5,-2,10,3.75}
L2={1,2,3,4,5}

CONCAT(L1,L2) = {2,5,-2,10,3.75,1,2,3,4,5}

CONCAT(L1,{5})▶L1 would add another element of value 5 onto the end of list **L1**, storing the resulting longer list back into **L1**.

ΔLIST(<list>)

This function produces a list which contains the differences between successive values in the supplied list. The resulting list has a length one less than the original.



Eg. L1={1,4,7,10,13}

ΔLIST(L1)={3,3,3,3}

MAKELIST(<expression>,<var_name>,<num>,<num>,<num>)

This function produces a list of the length specified using a rule of your choice. It is very useful, not only in programming but in statistical simulations and modeling.

The syntax is: **MAKELIST**(expression, variable name, start, end, increment)

where	*expression*	is the mathematical rule used to generate the numbers.
	variable name	is the variable that is to be used in the expression (any other letters will be taken constants).
	start	is the first value that *variable name* is be given.
	end	is the largest value that *variable name* is to take.
and	*increment*	is the amount that *variable name* should be incremented by.

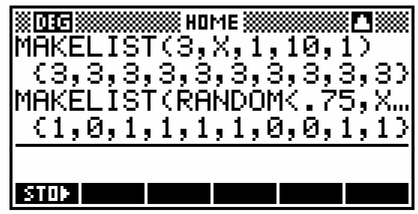
Examples are shown on the following page.

Eg. 1 **MAKELIST(X2,X,1,10,2)▶L1**
produces { 1, 9, 25, 49, 81 } as X goes from 1 to 3 to 5 to …
and also stores the result into **L1**.

Eg. 2 **MAKELIST(RANDOM,X,1,10,1)** produces a set of 10 random numbers.
The **X** in this case serves only as a counter since it does not appear in the expression.

Eg. 3 **MAKELIST(3,X,1,10,2)**
produces {3,3,3,3,3,3,3,3,3,3}.

The **MAKELIST** function can also be used to simulate observations on random variables.



For example, suppose we wish to simulate 10 Bernoulli trials with p = 0.75. We can use the fact that a test like X<4 or Y>0.2 returns a value of either 1 (if the test is true) or 0 (if the test is false).

Thus: **MAKELIST(RANDOM<0.75,X,1,10,1)**
will return a list of 1's and 0's corresponding to the simulated Bernoulli trials.

Various examples of this process are given in the Expert User section which immediately follows the chapter on Univariate Statistics on page 121.

πLIST(<list>)

This function returns the product of all the elements of a list.

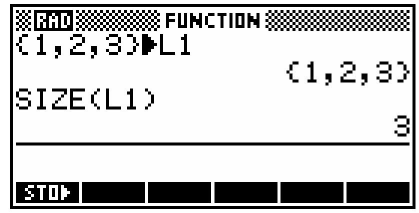
Eg. **πLIST({2,3,5})** would return a value of 30.

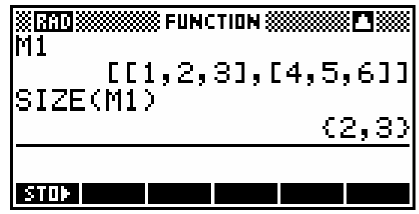
POS(<list>,<num>)

This function conducts a search of a list. It returns the position in the list of the first occurrence of the number you specify (see example right). It is of more use to programmers, who will not know in advance what a list contains.

If the number specified is not in the list it returns zero. If the value occurs in more than one place then only the first position is reported. The value specified can be either a number (as shown) or a variable or an expression to be evaluated.

SIZE(<list>) or SIZE(<matrix>)



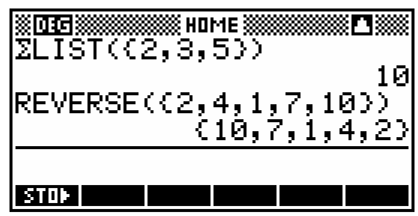


This function returns the size of the list or matrix specified. Since normal users would probably know anyway, and could find out easily via the list catalog, this is clearly another of those functions which are of more use to programmers (who won't know when they write their program just how long the list you will ask it to deal with will be when you eventually run the program). If the object is a matrix then the return value is a two element list as {rows, columns}.

ΣLIST(<list>)

This function returns the sum of all the elements of a list.

Eg. **ΣLIST({2,3,5})** would return a value of 10.

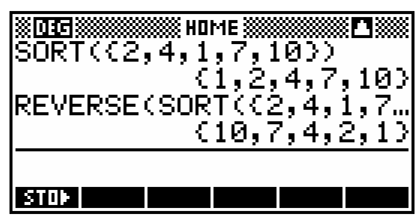


REVERSE(<list>)

This function reverses the order of elements in a list.

Eg. **REVERSE({2,4,1,7,10})** would return {10,7,1,4,2}

SORT({list})

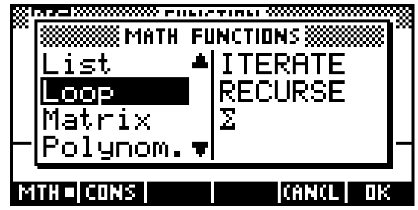


This function returns a list that is sorted into ascending order.

If you want the list in descending order instead then use **REVERSE(SORT({list}))**.

See page 137 for an example of using **SORT** to assign rank orders to a list of values.

The 'Loop' group of functions

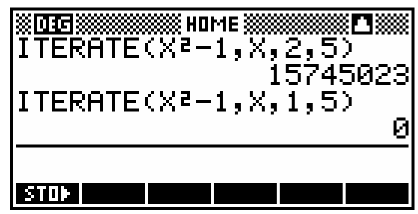


This is a group of functions that may be of use for students studying discrete functions and sequences but are primarily of use to programmers.

ITERATE(<expression>,<var_name>,<num>,<num>)

This function evaluates an expression in terms of a variable, starting with a supplied initial value, for a specified number of iterations. Each iteration uses the answer to the previous evaluation as the value for the variable in the next evaluation.

Eg. **ITERATE(X² - 1, X, 2, 5)** gives an answer of 15745023

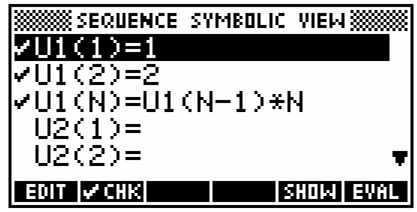


This answer is obtained as follows:

initial value:	X=2 $x^2 - 1$ = 3	first iteration using supplied initial value of 2.
new value:	X=3 $x^2 - 1$ = 8	second iteration
new value	X=8 $x^2 - 1$ = 63	third iteration
new value	X=63 $x^2 - 1$ = 3968	fourth iteration
new value	X=3968 $x^2 - 1$ = 15745023	fifth iteration

Final answer: X = 15 745 023

RECURSE



This functions is provided for programmers to let them define functions in the Sequence aplet.

For example, typing **RECURSE(U,U(N-1)*N,1,2) ▶U1(N)** seemingly produces no useful result in the **HOME** view, but would produce the result shown right in the **SYMB** view of the Sequence aplet. The resulting sequence is the factorial numbers.

The syntax is:

RECURSE(<seq.name>,<defn of term n>,<1st term>,<2nd term>)

and it must be stored into a sequence such as **U1,U2..U9,U0** for it to have any meaning.

Σ(<var_name>,<num>,<num>,<expression>)

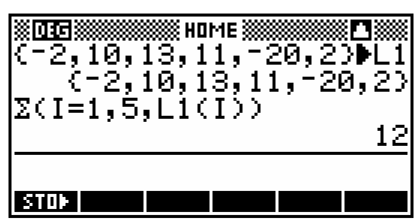
This function, also available on the keyboard, offers a way of calculating the results of summation notation problems. The syntax of the function is ordered in the same way as one reads a summation expression (see the examples below).

Eg. 1 $\sum_{i=1}^{5} i^2$ which expands to $1^2+2^2+3^2+4^2+5^2$ giving an answer of 55,

can be evaluated using **Σ(I=1,5,I²)**

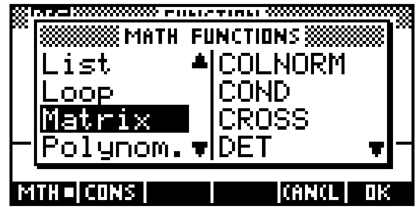
Eg. 2 $\sum_{i=1}^{6} x_i$ where $x_1=-2,\ x_2=10,\ x_3=13,\ x_4=11,\ x_5=-20,\ x_6=2$

can be evaluated by first defining a list **L1** as **{-2,10,13,11,-20,2}**
and then calculating **Σ(I=1,6,L1(I))**



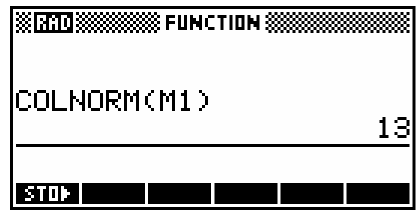
Note: Although the variable **I** was used as the summation index in each of the cases above, there is nothing special about them. 'i', 'j' & 'k' are simply the letters traditionally used in mathematical problems involving summation. When working in the hp 40gs CAS it is not possible to use 'i' because the CAS will interpret it as $\sqrt{i}$ rather than as a variable name.

The 'Matrix' group of functions



This group of functions is provided to deal with matrices.

> The scope of functions & abilities covered in this group is in fact vastly greater than would be required by the average high school student or teacher. In many cases supplying an explanation in more detail than the manual of what the function is used for would occupy many pages to no real useful gain. Consequently although some will include detailed examples, some of the functions will be covered only very briefly. Further examples are given in the chapter titled "Working with Matrices" on page 209.



COLNORM(<matrix>)

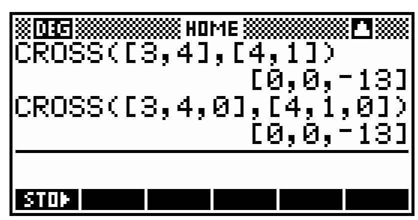
Finds the column norm of a matrix: the maximum, over all columns contained in the matrix, of the absolute values of the sum of the elements in each column.

Eg. For the matrix $M1 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ -1 & 5 & 4 \end{bmatrix}$, the column with the largest absolute sum of 13 is column 3.

COND(<matrix>)

Finds the condition number, the 1-norm (column norm), of a square matrix.

CROSS([vector],[vector])



This function finds the cross product of two vectors. Vectors for this function are written as single row matrices.

For example, $a = (3,4,0)$ or $\begin{pmatrix} 3 \\ 4 \\ 0 \end{pmatrix}$ would be written as **[3,4,0]**.

Note: If you want to use the **Matrix Catalog** to define your matrices for use with this function then you must use the **NEW** key to define them as real vectors rather than as matrices as the **CROSS** function is one of the few for which this matters.

DET(<matrix>)

This function finds the determinant of a square matrix. See page 213 for an example of its use in finding an inverse matrix.

Eg. If $A = \begin{bmatrix} 2 & 3 \\ -1 & 5 \end{bmatrix}$ then find det(A).

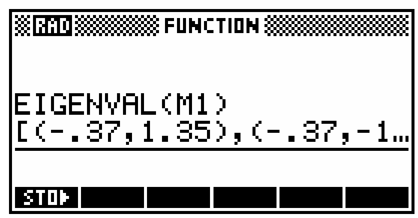
Ans: det (A) = 2x5-3x(-1)
= 13

See also: **INVERSE**, **RREF**

DOT([vector],[vector])

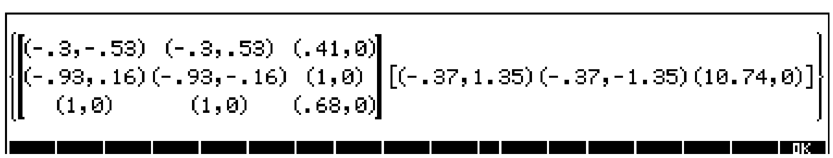
This function returns the dot product of two vectors. Vectors for this function are written as single row matrices. For example, $a = (3,4)$ or $\begin{pmatrix} 3 \\ 4 \end{pmatrix}$ would be written as **[3,4]**. See page 214 for a worked example.

EIGENVAL(<matrix>)



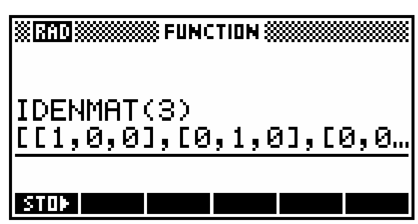
This function returns the Eigenvalues in vector form for a matrix. The values can be complex as shown right.

EIGENVV(<matrix>)



This function returns the Eigenvectors and Eigenvalues for a square matrix. It returns a list variable consisting of two arrays: a matrix containing the eigenvectors and a complex vector containing the eigenvalues.

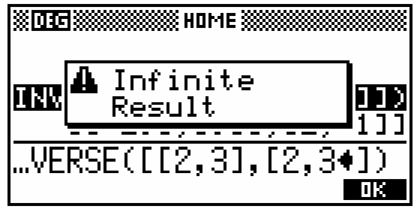
IDENTMAT(<size>)



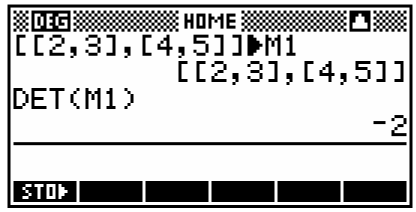
This function creates an n x n square matrix which is an identity matrix. For example, **IDENTMAT(4)** would produce a 4x4 identity matrix for use or storage. This is a function that is probably of more use to programmers than to the average user.

INVERSE(<matrix>)

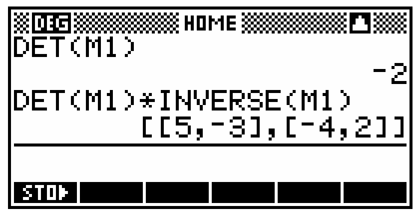
This function produces the inverse matrix of an n x n square matrix, where possible. A fully worked example of the use of an inverse matrix to solve a 3 by 3 system of equations is given in the chapter on Using Matrices on page 211 and in Appendix A on page 302.



An error message is given (see right) when the matrix is singular (det. zero).



Note: Some people write the inverse matrix as a fraction (one over the determinant) multiplied by a matrix, so as to avoid decimals and fractions within the inverse matrix. The calculator does not do this. If you want the matrix with the determinant factored out, then evaluate **DET(matrix)** first, record the fraction and then evaluate **DET(matrix) * INVERSE(matrix)** to obtain (usually) a non-fractional matrix.



i.e. $A = \begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix} \Rightarrow A^{-1} = \dfrac{1}{-2}\begin{bmatrix} 5 & -3 \\ -4 & 2 \end{bmatrix}$

Remember that the inverse matrix is not just the matrix, but the fraction times the matrix.

See also: **RREF**, **DET**

LQ(<matrix>)

This function takes an *mxn* matrix, factors it and returns a list containing three matrices which are (in order):

- an mxn lower trapezoidal matrix
- an nxn orthogonal matrix
- an mxm permutation matrix.

If you want to separate these matrices for later use then you should store them into a list variable.

For example, if **M1** was [[1,2,3],[4,5,6],[7,8,9]] then **LQ(M1)▶L1** would store the three resulting matrices into list variable **L1**. In the **HOME** view you could now enter **L1(1)▶M2** to store the first of the result matrices into **M2** and so on.

LSQ(<matrix1>,<matrix2>)

The least squares function displays the minimum norm least squares matrix (or vector).

LU(<matrix>)

This **LU** Decomposition function is similar to the **LQ** function on the previous page. It factors a square matrix into three matrices, returning them in the form of a list variable.

{[[lower triangular]],[[upper triangular]],[[permutation]]}

The upper triangular has ones on its diagonal. The matrices can be separated in the same method outlined for the **LQ** function.

MAKEMAT(<expression>,<rows>,<columns>)

The MAKEMAT function is used, mainly by programmers to manufacture a matrix with dimensions rows × columns, using the supplied expression to calculate each element.

Eg. **MAKEMAT(0,3,3)** returns a 3×3 zero matrix,

Note: If the expression contains the variables **I** and **J**, then the calculation for each element substitutes the current row number for **I** and the current column number for **J** during the calculation.

Eg. **MAKEMAT(I+J,3,3)** returns the matrix $\begin{bmatrix} 2 & 3 & 4 \\ 3 & 4 & 5 \\ 4 & 5 & 6 \end{bmatrix}$

QR(<matrix>)

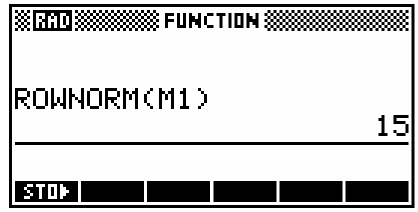
The **QR** function is similar to the **LQ** function on the previous page. It factors an m x n matrix into three matrices, returning them in the form of a list variable.

{[[m×m orthogonal]],[[m×n uppertrapezoidal]],[[n×n permutation]]}

RANK(<matrix>)

This function returns the rank of a rectangular matrix.

ROWNORM(<matrix>)



Finds the row norm of a matrix: the maximum, over all rows contained in the matrix, of the absolute values of the sum of the elements in each row.

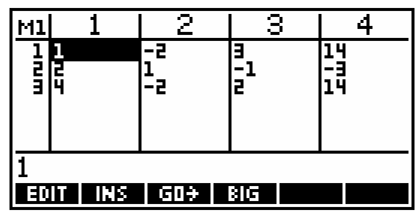
Eg. For the matrix $M1 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ -1 & 5 & 4 \end{bmatrix}$, the row with the largest absolute sum of 15 is row 2.

RREF(<matrix>)

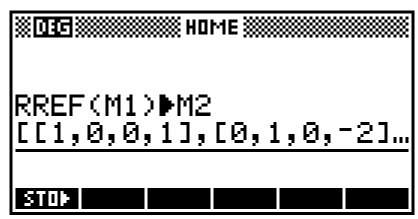
This function takes an augmented matrix of size n by n+1 and transforms it into reduced row echelon form, with the final column containing the solution.

Eg. The system of equations $\left.\begin{array}{l} x-2y+3z=14 \\ 2x+y-z=-3 \\ 4x-2y+2z=14 \end{array}\right\}$

is written as the augmented matrix $\left[\begin{array}{ccc|c} 1 & -2 & 3 & 14 \\ 2 & 1 & -1 & -3 \\ 4 & -2 & 2 & 14 \end{array}\right]$

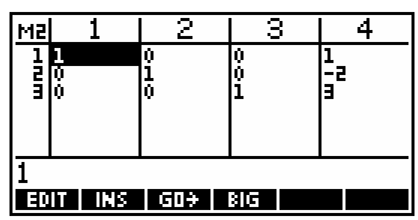


which is then stored as a 3x4 real matrix **M1**.



We now use the function **RREF** to change this to reduced row echelon form and store it as **M2**.

This gives the final result shown in the matrix **M2** on the right, giving a solution of (1, -2, 3).

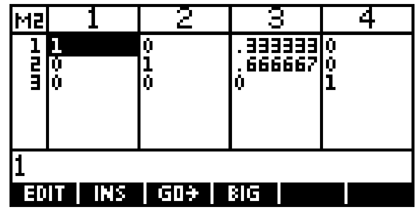


The huge advantage of this function is that it allows for inconsistent matrices which can't be solved by an inverse matrix.

For example, suppose we use the system of equations below, in which the third equation is a linear combination of the first two but the constant is not consistent with this - ie no solution.

$$\begin{cases} x+y+z=5 \\ 2x-y=-6 \\ 3y+2z=13 \end{cases}$$

If we solve this in the same way as before, the matrix which results is:



The final line of [0 0 0 1] indicates no solution. See the chapter "Working with Matrices" for more examples.

See also: **INVERSE**, **DET**

SCHUR(<matrix>)

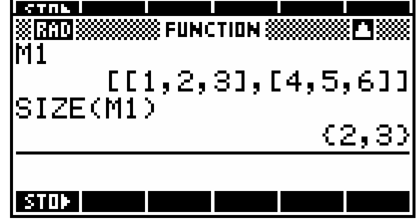
This function returns the Schur Decomposition for the square matrix supplied. The result is two matrices stored in a list. If the supplied matrix is real, then the result is:

{[[orthogonal]],[[upper-quasi triangular]]}.

If matrix is complex, then the result is:

{[[unitary]],[[upper-triangular]]}.

SIZE(<list>) or SIZE(<matrix>)



This function returns the size of the list or matrix specified. Since normal users would probably know anyway, and could find out easily via the list catalog, this is clearly another of those functions which are of more use to programmers (who won't know when they write their program just how long the list you will ask it to deal with will be when you eventually run the program). If the object is a matrix then the return value is a two element list as {rows, columns}.

SPECNORM(<matrix>)

This function returns the Spectral norm of a matrix.

SPECRAD(<matrix>)

This function returns the Spectral radius of a matrix.

SVD(<matrix>)

This function performs a Singular Value Decomposition on an m × n matrix. The result is two matrices and a vector:

{[[m × m square orthogonal]],[[n × n square orthogonal]],[real]}.

SVL(<matrix>)

This function returns a vector containing the singular values of the supplied matrix.

TRACE(<matrix>)

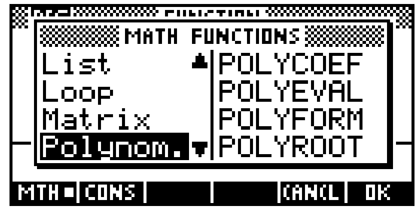
This function finds the trace of a square matrix. The trace is equal to the sum of the diagonal elements or the sum of the eigenvalues.

TRN(matrix)

This function returns the transpose of an n x m matrix.

For example, if $M1 = \begin{bmatrix} 2 & 3 \\ 1 & -2 \\ 0 & 4 \end{bmatrix}$ then TRN(M1) would return $\begin{bmatrix} 2 & 1 & 0 \\ 3 & -2 & 4 \end{bmatrix}$.

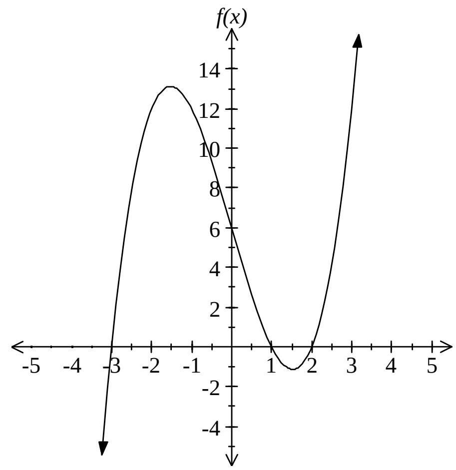
The 'Polynomial' group of functions



This group of functions is provided to manipulate polynomials.

We will use the function shown right to illustrate some of the tools in the Polynomial group. Its equation is:

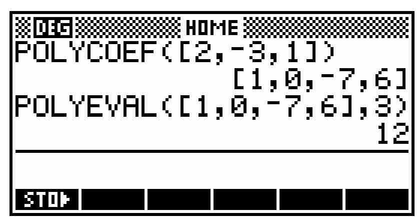
$$f(x)=(x-2)(x+3)(x-1)=x^3-7x+6$$



POLYCOEF([root1,root2,…])

This function returns the coefficients of a polynomial with roots $x_1, x_2, x_3, \ldots$ The roots must be supplied in vector form means in square brackets.

The function $f(x)$ above has roots 2, -3 and 1. The screen shot below shows **POLYCOEF** correctly giving the coefficients as 1, 0, -7 and 6 for a final polynomial of $f(x)=x^3-7x+6$.

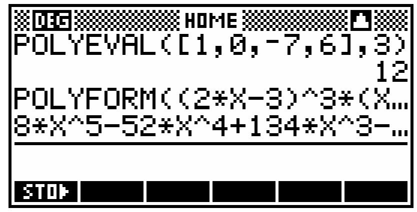


POLYEVAL([coeff1,coeff2,…],value)

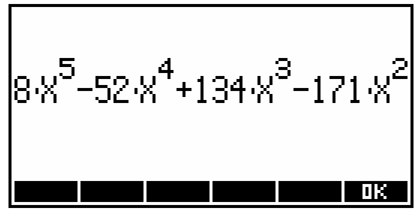
This function evaluates a polynomial with specified coefficients at the point specified. The coefficients must be in square brackets, followed by the value of x (not in brackets). ie. $f(x)=x^3-7x+6$ has value 12 at x = 3.

Note: This is a function that is aimed more at programmers. For normal users it is probably more efficient to enter the function into the **SYMB** view of the Function aplet and then either use the **NUM** view to find the function values required, or simply type **F1(3)**, **F1(-2)** etc. in the **HOME** view.

POLYFORM(<expression>,<var_name>)



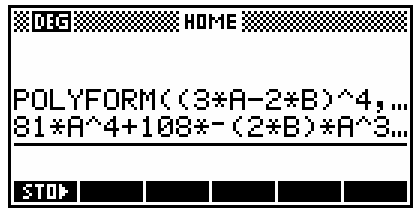
This is a very powerful and useful polynomial function. It allows algebraic manipulation and expansion of an expression into a polynomial. The expected parameters for the function are firstly the expression to be expanded, and secondly the variable which is to be the subject of the resulting polynomial. If the expression contains more than one variable then any others are treated as constants.



Eg. 1 Expand $(2x-3)^3(x-1)^2$

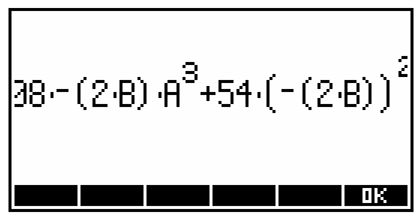
Result: $8x^5-52x^4+134x^3-171x^2+108x-27$

The resulting polynomial is shown both as it appears in the **HOME** view and as it appears after pressing the **SHOW** key. Once it appears in the **SHOW** window, of course, it can be scrolled right and left to see the missing terms.



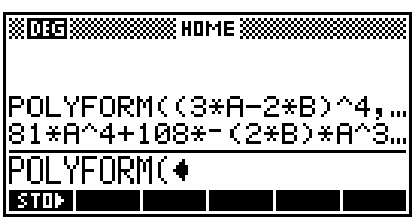
Eg. 2 Expand $(3a-2b)^4$

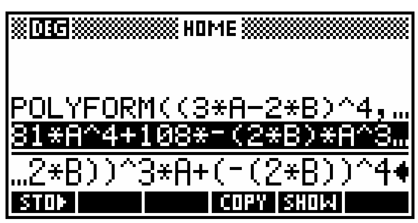
This function contains two variables, **A** and **B**, which must be expanded separately.

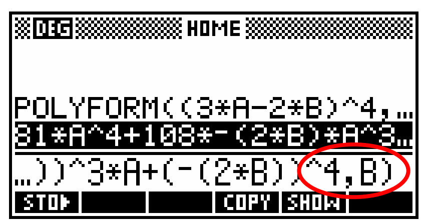


The first expansion, treating **A** as the variable, is done using the expression **POLYFORM((3A-2B)^4,A)**. As you can see if you examine the view after pressing **SHOW**, the expansion of the expression in terms of **A** has been done, but the terms involving **B** are not fully evaluated.

The solution to this is to use **POLYFORM** again. Use the **MATH** menu to fetch the **POLYFORM** function to the edit line, then move the cursor up to the partially evaluated expression that was the result of the previous **POLYFORM**. Copy it into the edit line and add a comma, a **B** and an end bracket. Pressing **ENTER** will now evaluate the terms involving **B**.







After pressing **ENTER** the for the second evaluation, the result is shown right (after pressing **SHOW**).

POLYROOT([coeff1,coeff2,…])

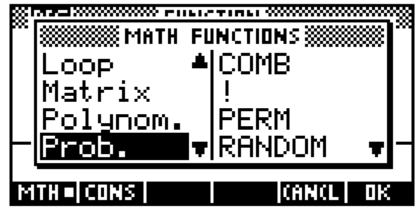
This function returns the roots of the polynomial whose coefficients are specified. The coefficients must be input as a vector in square brackets.

Eg. Using our earlier function of $f(x)=(x-2)(x+3)(x-1)=x^3-7x+6$ we can enter the coefficients as [1, 0, -7, 6].

As you can see in the screen shot, the roots of 2, -3 and 1 have been correctly found.

See page 309 for details on finding roots of real and complex polynomials using the CAS on the hp 40gs.

The 'Probability' group of functions



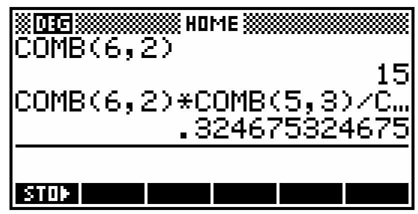
This group of functions is provided to manipulate and evaluate probabilities and probability distribution functions (p.d.f.'s).

COMB(<n>,<r>)

This function gives the value of ${}^{n}C_{r}$ using the formula ${}^{n}C_{r} = \dfrac{n!}{(n-r)!r!}$.

Eg. Find the probability of choosing 2 men and 3 women for a committee of 5 people from a pool of 6 men and 5 women.

$$p = \frac{\binom{6}{2}\binom{5}{3}}{\binom{11}{5}} = 0.3247$$



Notes:

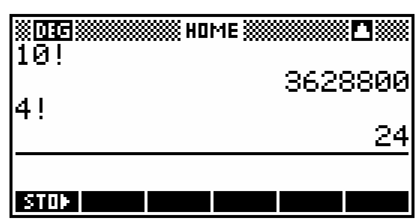
1. The reason for the single '**COMB(6,2)**' above the main calculation is to save time. Rather than using the **MATH** menu for every entry of the **COMB** function, you can enter it once and then **COPY** it repeatedly, changing the parameters each time.

2. For large values of **N** (>150 or so) it is important to use the **COMB** function rather than using **N!/((N-R)!*R!)** because the massive values involved in calculating the factorials will cause inaccuracies internally. The **COMB** function has internal methods built in which avoid this and give accurate answers.

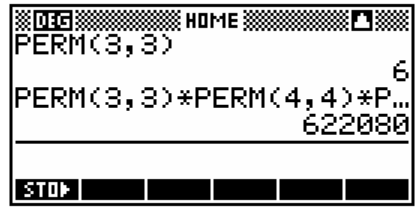
3. If you are <u>solving</u> for the value of **N**, such as in a Binomial probability calculation then you must use **N!/((N-R)!*R!)** rather than **COMB**.

The ! function

The factorial function finds the number of possible permutations of an entire collection of *n* objects.



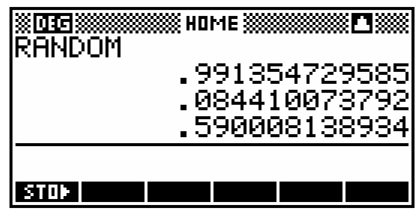
PERM(<n>,<r>)



This function gives the value of ${}^{n}P_{r}$ using the formula ${}^{n}P_{r} = \dfrac{n!}{(n-r)!}$.

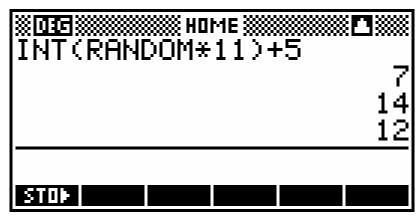
Eg. How many ways can 3 Math, 4 English, and 6 German books be arranged on a shelf if all the books from a subject must be together?
Ans: $\left({}^{3}P_{3} \times {}^{4}P_{4} \times {}^{6}P_{6}\right) \times 3!$ = 622080

RANDOM



This function supplies a random 12 digit number between zero and one. If you want a series of random numbers, just keep pressing **ENTER** after the first one.

Eg. Produce a set of random integers between 5 and 15 inclusive.



Use the expression **INT(RANDOM*11)+5**
The **RANDOM*11** produces a range from **0** to **10.999999**. This is then dropped down to the integer below by the **INT** function, giving a range of integers **0,1,2,3,...,10**. The final adding of **5** gives the correct range.

A similar command of **INT(RANDOM*6)+1)** will produce simulated rolls of a normal die.

RANDSEED(<number>)

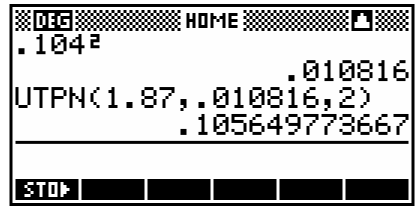
It is important to realize that the values produced by the **RANDOM** command are not truly random numbers. Inside the calculator is a mathematical procedure (an algorithm) which uses a 'seed' number to produce them. Unfortunately, when taken straight out of the box, two calculators will produce exactly the same sequence of "random" numbers! This can be a problem. For example, a class set of calculators doing a simulation of dice might all produce the same numbers.

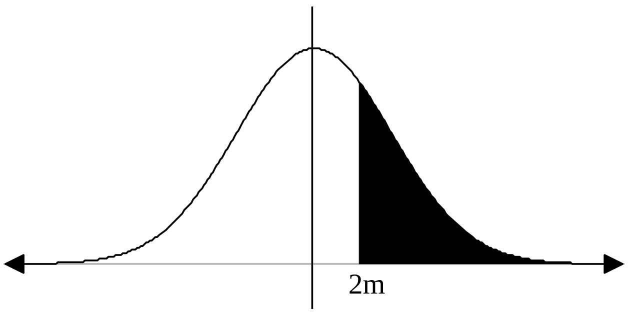
The simple solution is to seed the random number algorithm with different values for each machine. This is done using the function **RANDSEED(...)** and, providing each person uses a different number, it should only be necessary to do it once (unless the calculator is reset). The simplest method is to use **RANDSEED(Time)** since the value of **Time** will be different on each calculator. Since the same seed will produce the same sequence of numbers, this can be convenient for teachers in test situations.

UTPN(<mean>,<variance>,<value>)

This function, the 'Upper-Tail Probability (Normal)', gives the probability that a normal random variable is greater than or equal to the value supplied. Note that the variance must be supplied, NOT the standard deviation.

Eg. 1. Find the probability that a randomly chosen individual is more than 2 meters tall if the population has a mean height of 1.87m and a standard deviation of 10.4cm



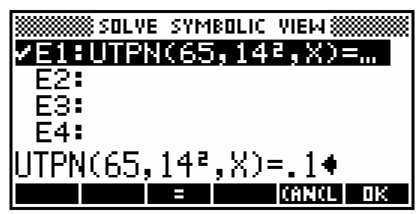


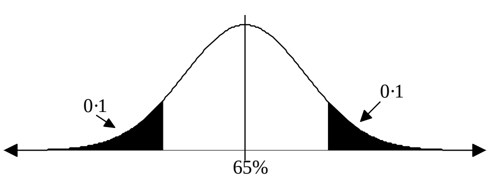
$$\bar{x} = 1.87m, \sigma = 0.104m$$

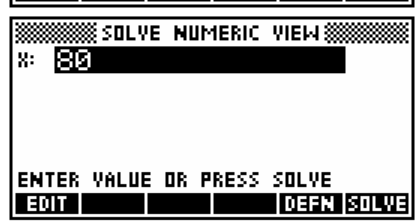
$$\Rightarrow \sigma^2 = 0.010816$$

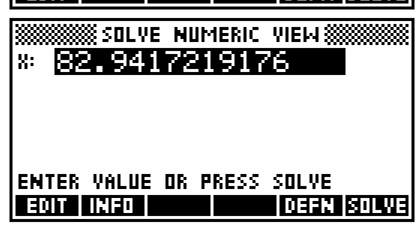
Ans: P(height>2m) = 0.1056

Eg. 2. The population of Year 12 Applicable Mathematics students had a mean exam score of 65% and a standard deviation of 14%. What two scores will cut off the top and bottom 10% of students?









i.e. Find x_0 such that $P(x > x_0) = 0 \cdot 1$

Using the Solve aplet (right) we can reverse the normal direction of the **UTPN** function.

Enter the expression to be solved for into the **SYMB** view as shown above, then switch to the numeric view. Enter a guess of 0.8 (80%) and then press **SOLVE**.

The second value can be found by using the symmetry properties of the Normal Distribution, but it is probably just as fast to go back to the **SYMB** view, change the 0.1 to 0.9 and then re-**SOLVE**. Remember that an **EDIT** key is provided in the **SYMB** view to allow you to change the expression without having to retype it.

Final answer… 47.06% and 82.94% are the cut-offs.

> **Calculator Tip**
> The normal order for the arguments in the **UTPN** function is **UTPN(mean, variance, value)** and this results in the upper-tailed probability. However, many textbooks work with the lower-tailed probability instead. Fortunately the function can easily be adapted for this.
>
> If you instead enter the function's parameters as **UTPN(value, variance, mean)**, reversing the normal positions of *value* and *mean*, then the probability given will be the lower-tailed value. The reason for this lies in the symmetry properties of the normal curve.

UTPC(<degrees>,<value>)

This is the Upper-Tailed Chi-Squared probability function. It returns the probability that a χ^2 distribution with the supplied number of degrees of freedom is greater than the value supplied. See page 147 for an example of this function's use.

UTPF(<numerator>,<denominator>,<value>)

This is the Upper-Tailed Snedecor's F probability function. It returns the probability that a Snedecor's F distribution with numerator degrees of freedom (and denominator degrees of freedom in the F distribution) is greater than the supplied value.

UTPT(<degrees>,<value>)

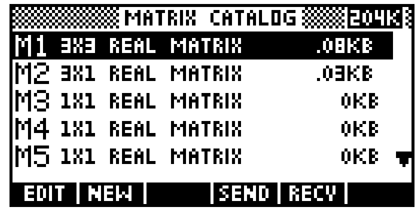
This is the Upper-Tailed Student's t probability function. It returns the probability that a Student's t distribution with the supplied number of degrees of freedom is greater than the supplied value.

30

WORKING WITH MATRICES

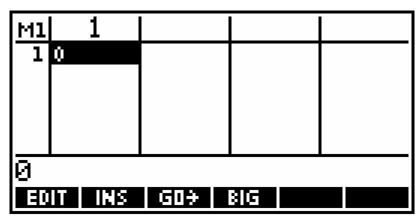
The hp 39gs & hp 40gs deal very well with matrices. They offer many powerful tools as well as a special **MATRIX Catalog** with full editing facilities.

The **MATRIX Catalog** is entered by pressing **MATRIX** (located above the 4). It allows storage of ten matrices (**M1,M2,..M9,M0**) which can be any size, depending only on available memory. In the example shown right, the catalogue contains two matrices, a 3 x 3 and a 3 x 1. The reason that the catalogue specifies that they are 'real matrices' is that the calculator is capable of storing and manipulating not only matrices of real numbers but also matrices of real vectors, complex numbers and complex vectors. Some functions in the **MATH** menu are specifically aimed at matrices, others at vectors. The **NEW** key pops up the menu shown on the right, replacing the highlighted matrix with any empty one of the new type that you specify. Press **SHIFT CLEAR** to delete all.

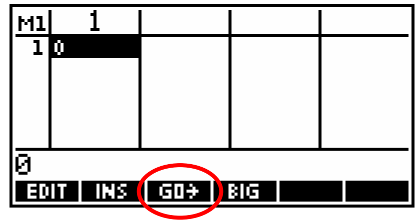


Matrices, like most other objects on the calculator, can be sent to and received from a computer or another calculator using the **SEND** and **RECV** keys. On the hp 40gs this is done using the two cables supplied, one being a mini-USB for use with a PC and the other being a mini-serial for inter-calculator communication. On the hp 39gs only the mini-USB cable is supplied for use with the PC since inter-calculator communication is done using the calculator's infra-red link.

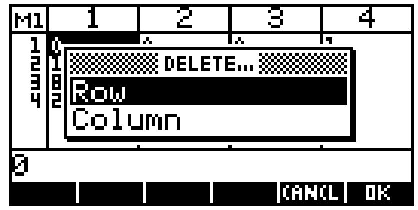
We'll begin by entering a matrix into the catalogue to practice simple editing. If there are any matrices currently in the catalogue, use **SHIFT CLEAR** to delete the whole catalogue. Move the highlight to matrix **M1** and press the **EDIT** key. The normal state for a blank matrix is to contain nothing but a single zero, which is why they all register as 1x1 even after erasure.



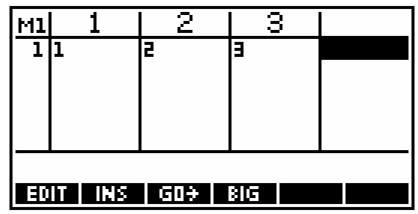
If you look at the list of screen keys on the bottom of the view, you will see one labeled **GO→**. This determines which way the highlight will move (across or down) when you enter a number. If you press the key repeatedly you will see it change from **GO→** (across), to **GO↓** (down), to **GO** (no movement).



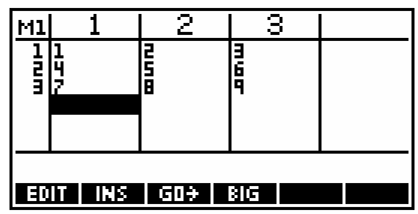
You will also see the usual **BIG** key, and an **INS** key that can be used to insert an extra row or column into an existing matrix. The keyboard **DEL** key can be used to delete a row or column.



With movement set to **GO→**, type in the numbers 1, 2 and 3, pressing **ENTER** after each. Your display should now look like the screen shot right.



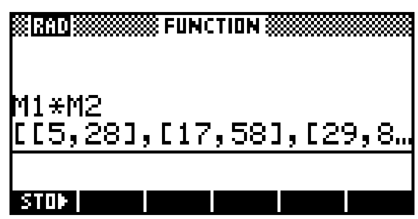
If you now press the down arrow key on the keyboard the highlight will move back to the first element in the second row. Enter the numbers 4, 5 and 6 and you will find that the calculator automatically drops down to row three without the need to use the down arrow key again, since it now knows how many columns the matrix is to contain. Finish your matrix so that it looks like the one shown right.

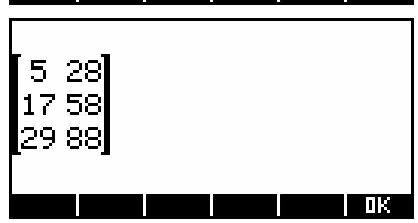


Now press **MATRIX** to switch back to the **MATRIX Catalog** and, with the highlight on **M2**, press **EDIT** and create the matrix shown right.

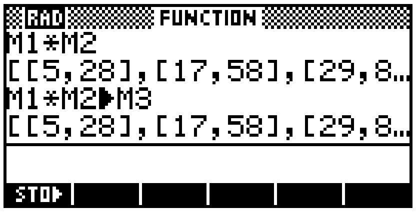
$$\begin{bmatrix} 2 & -1 \\ 3 & 4 \\ -1 & 7 \end{bmatrix}$$

Switch to the **HOME** view and multiply **M1*M2**. As you can see, the result displayed in the **HOME** view is not very useful or readable. Highlighting it and pressing **SHOW** displays the matrix in a far more readable manner.

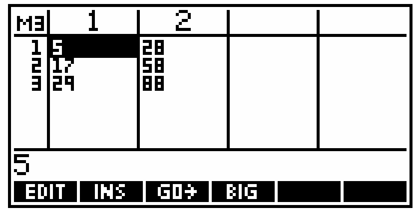




Another method is to store the result into a third matrix and then to view it through the Edit screen of the **MATRIX Catalog**. This is shown below.



Matrix **M3** is created left and edited right.



Probably the most common functions that you will use are **INVERSE, DET** and **TRN** (transpose), so some worked examples are included which use them. There are also a number of further worked examples involving matrices in the section at the back of the book.

Eg. 1 Solve the system of equations:

$$\begin{cases} 2x+3y-z=-6 \\ x-3y+z=12 \\ 3x-y+4z=13 \end{cases}$$

Solution: The system of equations can be represented as the system of matrices:

$$\begin{bmatrix} 2 & 3 & -1 \\ 1 & -3 & 1 \\ 3 & -1 & 4 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \end{bmatrix}=\begin{bmatrix} -6 \\ 12 \\ 13 \end{bmatrix}$$

and this system can then be algebraically rearranged to:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}=\begin{bmatrix} 2 & 3 & -1 \\ 1 & -3 & 1 \\ 3 & -1 & 4 \end{bmatrix}^{-1}\begin{bmatrix} -6 \\ 12 \\ 13 \end{bmatrix}$$

where the inverse matrix is…

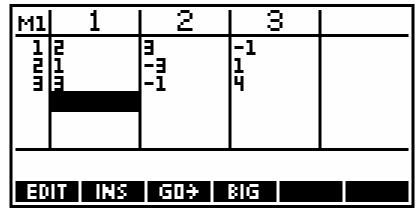
$$\begin{bmatrix} 2 & 3 & -1 \\ 1 & -3 & 1 \\ 3 & -1 & 4 \end{bmatrix}^{-1}$$

which gives a final answer of $\begin{bmatrix} x \\ y \\ z \end{bmatrix}=\begin{bmatrix} 2 \\ -3 \\ 1 \end{bmatrix}$

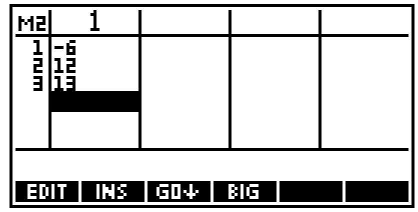
The method for doing this on the calculator is as follows…

Step 1. Enter the **MATRIX Catalogue**. Use **SHIFT CLEAR** to erase all matrices if desirable.

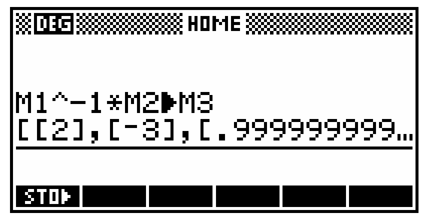
Step 2. Enter the 3x3 matrix of coefficients in **M1**.



Step 3. Enter the 3x1 matrix of into **M2**.
Note the change to **GO↓** in order to make entering numbers easier.



Step 4. Change to the **HOME** view, evaluate $A^{-1} \times b$ using any of the following three methods (all of which are acceptable to the hp 39gs or hp 40gs), and store the result into **M3**.



(a) **M1^{-1}*M2**

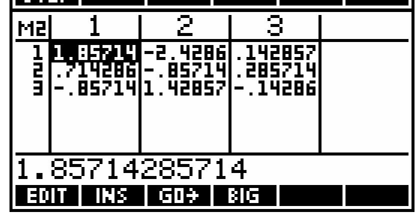
(b) **M2/M1**

(c) **INVERSE(M1)*M2**

The best of these is probably the first. The inverse **x^{-1}** function is on the keyboard and this makes it more convenient to use. The only problem with the second method is that it does not make it clear that the operation is really left-multiplication by the inverse rather than division, an operation which is not strictly speaking defined for matrices but is allowed for convenience on the calculator. The third option is actually identical to the first. The original hp 38g only had the **INVERSE** function and the **x^{-1}** function was added in later models. They are mathematically identical but the function **x^{-1}** can be found on the keyboard and so is simpler to use.

The answer displayed is **[[2],[-3],[.999999999999]].** This is really the same as x = 2, y = -3, z = 1. The strange answer for z = 1 is caused by internal precision errors. You are expected to realize yourself that it should be z = 1.

Eg. 2 Find the inverse matrix A^{-1} for the matrix $A = \begin{bmatrix} 2 & 1 & 4 \\ 1 & 1 & 3 \\ -2 & 4 & -1 \end{bmatrix}$

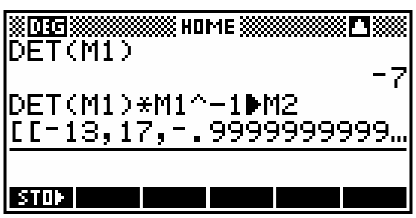


The first step is to store the matrix A into **M1**. If you now simply store its inverse into **M2** you will find, depending on the determinant, that the result is probably a collection of decimal values (see right).

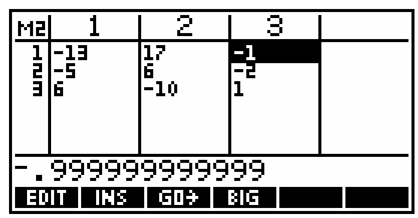
This answer is correct and we could stop there. However, this is not the best way to display the answer. The fact that the determinant is incorporated into the inverse makes whole numbers unlikely.

A better way is often to write the answer as a fraction $\frac{1}{\det(A)}$ multiplied by a matrix of whole numbers. If we multiply the inverse by the determinant then we can usually remove the fractions.

i.e.

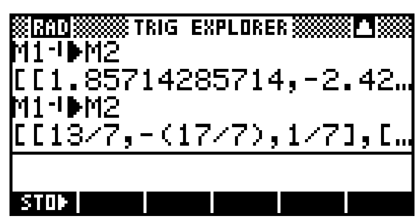


with **M2** being…

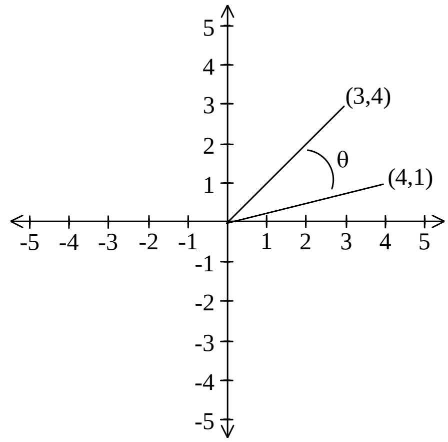


Thus we can finally write: $A = \frac{1}{-7}\begin{bmatrix} -13 & 17 & -1 \\ -5 & 6 & -2 \\ 6 & -10 & 1 \end{bmatrix}$

An alternative to this is to change into *Fraction 6* numeric format in the **MODES** view. This will give a matrix of fractional values which can be seen more clearly using **SHOW**.



Eg. 4 Find the angle between the vectors $a = (3,4)$ and $b = (4,1)$.



Using the formula that

$$\underline{a} \bullet \underline{b} = |\underline{a}|.|\underline{b}|.\cos\theta$$

where $\underline{a} \bullet \underline{b}$ is the dot product,
we can rearrange to obtain:

$$\cos\theta = \frac{\underline{a} \bullet \underline{b}}{|\underline{a}|.|\underline{b}|}$$

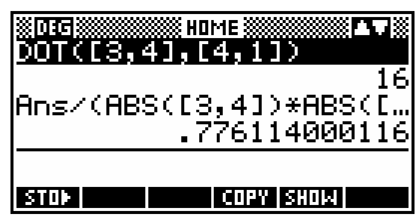
This substitutes to give a solution of:

$$\cos\theta = \frac{(3,4) \bullet (4,1)}{|(3,4)|.|(4,1)|}$$

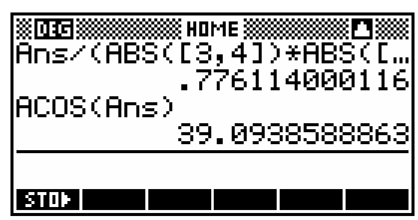
$$= \frac{3 \times 4 + 4 \times 1}{\sqrt{3^2 + 4^2}.\sqrt{4^2 + 1^2}}$$

$$= \frac{16}{5\sqrt{17}}$$

$$\theta = 39 \cdot 09^\circ$$



On the calculator, the functions **DOT** and **ABS** give the dot product and magnitude respectively, *when fed with vectors.* The calculator writes vectors as row matrices.
For example $a = (3,4)$ would be written as **[3,4]**.



The calculations are shown in the two screen shots on the right.
Remember to change into degree mode first.

The list of matrix functions available through the **MATH** menu is covered starting on page 195. Not all functions are covered, since many of them go far beyond the requirements of the average high school student at whom this book is aimed.

31

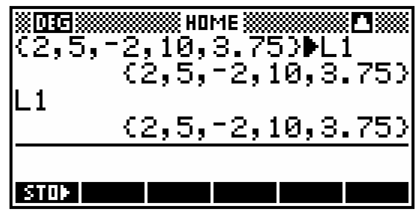
WORKING WITH LISTS

A list in the hp 39gs or hp 40gs is the equivalent of a mathematical set. As with a set, it is written as numbers separated by commas and enclosed with curly brackets.

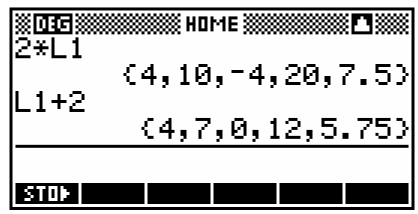
Eg. {2,5,-2,10,3.75}

Using the **HOME** view these lists can be stored in special list variables. There are ten of these **L1,L2,..L9,L0.**

Eg. {2,5,-2,10,3.75}▶L1

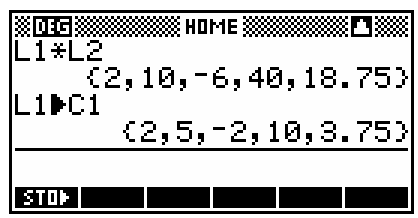


Typing **L1** and then **ENTER** will then retrieve the list. Lists can also be multiplied by a constant and have a constant added to them (see below)

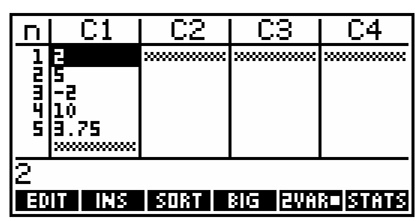


If we store another list of the same length into **L2**, then the two lists can be multiplied together. The resulting list is obtained by multiplying each element in the first list by the matching element in the second list.

Many of the normal mathematical functions also work on lists of numbers by performing the operation on each individual element. Lists on the calculator can contain more than simply numbers. For example, the return value for some matrix functions is a list where each element is a matrix. Elements of a list can be matrices, lists and many other things.

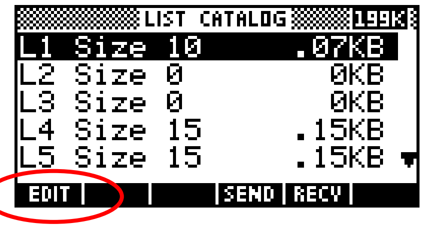


The column variables **C1,C2..C9,C0** in the Statistics aplet are actually list variables attached to their aplet and can be used as extra storage if you need more list variables. The statistical variables have the additional advantage, of course, that they can be graphed in the Statistics aplet and with all the usual statistical analysis measures available. To transfer a list variable to a statistics variable, just store one into the other (see right). As you can see in the second view, the list has been transferred to **C1**. Pressing **STATS** would now give the usual statistical measures for the newly created column.

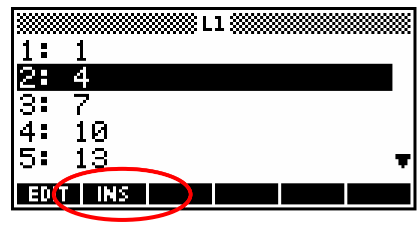


There are also a number of special functions available for list variables which are contained in the *List* group of functions in the **MATH** menu. See page 190.

A special **List Catalog** is provided which allows easier entering and editing of lists. If you look above the 7 key you will see a label of **LIST** which gives you access to this catalog. When you enter this catalog you will see the screen on your right.



Pressing the **DEL** key when on one of these lists will delete that list, clearing the memory it uses. Pressing **BLUE CLEAR** (above the **DEL** key) will clear all lists. In both cases a query message pops up asking whether you are sure you want to do this.

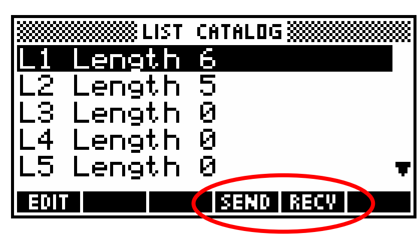


If you press the **EDIT** key then you will enter a special editing screen which allows you to change individual values, delete values (using the **DEL** key) and insert new elements into a list. If you press the **INS** key with the highlight in the position shown, then a zero is inserted before the 4, with all the elements below shifted down one position. You can then type your new number.

Having entered the editing screen, you may be wondering how to exit, since no key is apparently provided for this. Simply press **SHIFT LIST** again to return to the catalog level (or press **HOME** if you've finished altogether).

Changing of individual elements of a list can also be done in the **HOME** view, if not quite so easily. As we saw earlier, typing **L1** and pressing **ENTER** will display an entire list in the **HOME** view. If you want to see just element 3, for example, then typing **L1(3)** will display just that one element (see right). As you can also see, you can change just one element using the store command. It is not possible to insert an extra element, except in that you could add an element to the end of a list using the **CONCAT** function discussed on the next page.

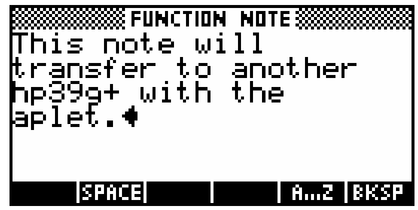
Lists can be sent from one calculator to another using the infra-red link on the hp 39gs or using the supplied mini-serial cable with the hp 40gs. The procedure is the same as that for sending aplets from one calculator to another.



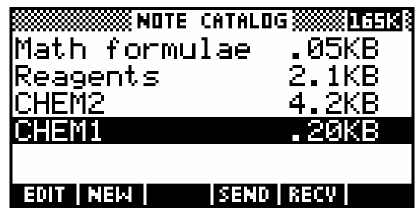
32

WORKING WITH NOTES & THE NOTEPAD

The hp 39gs & hp 40gs provide access to Notes which can either be attached to an aplet or exist independently. The notes belonging to the standard aplets are blank unless you add to them, but copies you transfer from a computer or another calculator may have had notes added to them as instructions on how to use them. In particular, any special aplets you download to your calculator from the internet may have instructions as a note and/or perhaps a sketch.

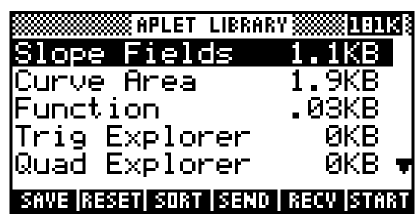


Every aplet has a note attached which, by default, is blank. This note is available via a key labeled **NOTE** (above **APLET**). Thus an instruction to 'press **NOTE**' will refer to the current aplet's Note.



In addition to this, there is a **Notepad Catalog** containing independent notes which is available via a **NOTEPAD**, found above zero. In the explanations below, an instruction to 'press **NOTEPAD**' will always refer to the **Notepad Catalog**, not to the aplet Note. The average user will probably not use the aplet Note.

Notes attached to aplets are explained in the chapter "Using, Copying & Creating Aplets" on page 226, so only a brief example is given below.

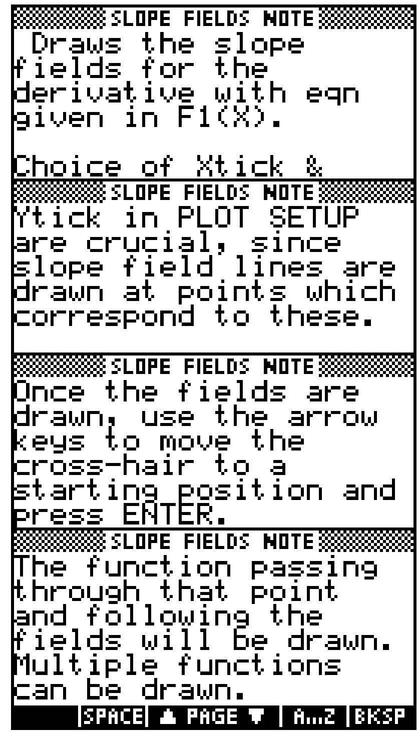


The Slope Fields aplet is a special educational aplet that can be used to introduce calculus students to the concept of fields of slopes. It has been downloaded onto my calculator from the author's web site the ***HP HOME view*** web site (http://www.hphomeview.com) and is now found on the **APLET** key.

This is not a standard aplet. Unless you download this aplet yourself you will not find it on your calculator!

When this aplet is run, the first thing visible is the **VIEWS** menu. This menu is set up by the programmer to control the aplet
(see "Programming the hp 39gs & hp 40gs" on page 255).

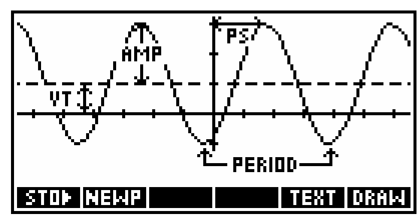
If you **CANCL** the menu and press **NOTE** on this particular aplet then you would see the attached Note shown right. This is quite common with downloaded aplets. Since they are non-standard, the author often ensures you have some instructions, although most of the documentation is generally in a Word® or PDF® document that comes with the aplet.



This particular set of notes consists of a number of pages, but most aplet Notes are not that extensive. Indeed there may not be one if the operation of the aplet is fairly obvious.

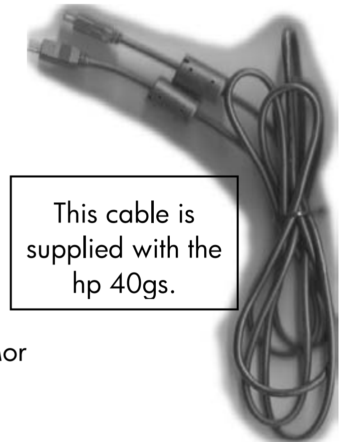
Most people don't read notes or explanations so it is debatable how much use they are anyway.

Some aplets also have sketches attached to them but this consumes approximately 1Kb of memory per sketch and so was not common on earlier models which only had 23Kb of RAM memory and so didn't have a lot to spare. On the hp 39gs & hp 40gs, as programmers become used to having plenty of memory to use, aplets have become larger and more powerful.



An example is shown above right of a sketch from one of the aplets available from HP's web site called 'Periodic'. If you installed this aplet on your calculator then this sketch could be viewed via the **SKETCH** key above the **MATH** key. Sketches can have multiple pages and are covered on page 222.

Independent Notes and the Notepad Catalog



This cable is supplied with the hp 40gs.

Most users are far more concerned with the **Notepad Catalog**. Notes held in it are independent of any aplet rather than being attached to only one.

These notes can be sent to (received from) another calculator or from a computer via the **SEND** and **RECV** keys in the **Notepad Catalog**. On the hp 39gs this is done via the infra-red link. On the hp 40gs there is no infra-red capacity and the transfer is done via the supplied mini-serial cable shown right. Mor on cables and their use can be found on page 237.

Since the hp 39gs is the more widely distributed model we will mainly discuss infra-red communication. The method is basically the same when using the mini-serial cable supplied with the hp 40gs.

The **Notepad Catalog** holds text material. For example Physics information or your mathematics homework or notes on trig graphs. As an exercise, we will develop a Note containing some common formulas.

> What notes you are allowed to take into tests is a matter for your school or university's policy. You should not cheat by taking banned material into a test or examination. It is wrong and could result in severe penalties. Some institutions require a complete reset before any examination. See page 42 on how to do this.

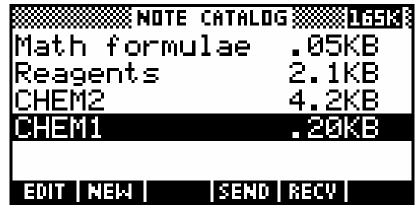
Notes can be shared between students in the same way as can be done for aplets, lists, matrices and programs. It is worth pointing out that this will not help you in a test situation since, on the hp 39gs the strength of the infra-red link is such that it will only operate over extremely short distances and on the hp 40gs the cable is going to be just a little obvious!

Notes also consume memory. Even on the hp 39gs or hp 40gs, if you accumulate enough Notes you will eventually find that you run out of space. See page 30 for information on the memory.

In addition to this, the calculator was never designed nor intended to be a typewriter. Inputting text is a slow process. As you begin to become more familiar with the positions on the keyboard where the various letters of the alphabet can be found your typing speed will improve, but it will never remotely compare to typing on a keyboard.

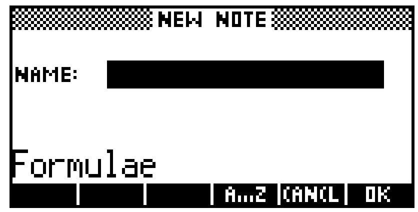
If you want to prepare anything more than small Notes then software is available to allow you to edit Notes on a computer. See page 252. For the previous models, the hp 38g, hp 39g and hp 40g, it was necessary to purchase a special cable to connect to the serial port on a PC. For the hp 39g+, hp 39gs and hp 40gs a cable is supplied with the calculator to allow you to connect to the USB port.

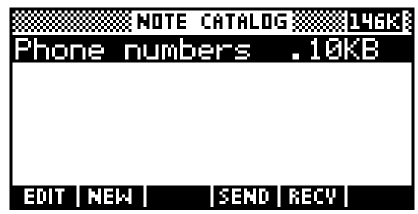
Creating a Note



Let's create a small Note containing some commonly used formulas. Press **SHIFT NOTEPAD** (not **SHIFT NOTE**) and you will see the **Notepad Catalog** shown right. Yours will probably be empty.

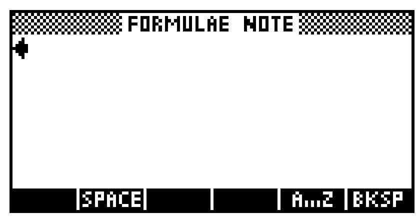
The keys at the bottom of the screen allow you to **EDIT** an existing Note, create a **NEW** one or to **SEND** and **RECV** Notes to or from another calculator (or a computer). A Note is deleted using the **DEL** key, while the **SHIFT CLEAR** key will delete all Notes in the catalog.



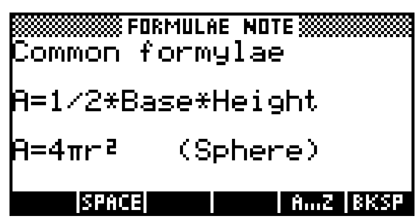


Press the **NEW** key to begin a fresh Note. You will be presented with the screen on the right, requesting that you enter a name for the Note. Names can be any length and can contain any characters, alphabetic, numeric, spaces as well as any from the **CHARS** menu. However, if the name is longer than 14 characters it will be truncated in the display because of the space taken up by the indication of how much memory is used. When you press **ENTER** after typing in the name, you will see a blank screen waiting for you to begin typing in your Note.

Before we begin, a word about making your typing easier. The normal method to enter an alphabetical character is to use the **ALPHA** key. Lowercase letters are obtained by pressing **SHIFT** first. The **ALPHA** key can be held down with one hand while you type with the other, but this isn't convenient and lowercase can't be locked this way. Space can be obtained via **ALPHA SPACE**, with **SPACE** located above the + key.



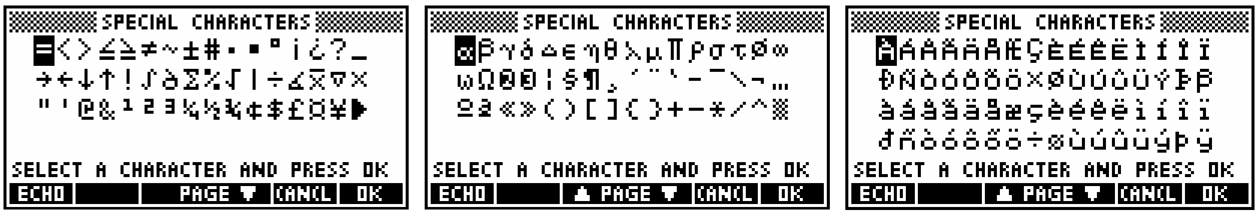
At the bottom of the screen in the Notepad view you will see screen keys labeled **A...Z**, **SPACE** and **BKSP** (backspace). If you press **A...Z** then all the keys from then on will type their alphabetic values rather than their normal mathematical functions. Pressing the **SHIFT** key first before the **A...Z** key will lock it into lowercase rather than into uppercase. Pressing the **A...Z** key again disengages it. The **SPACE** key is there purely for your convenience since it is also on the keyboard.



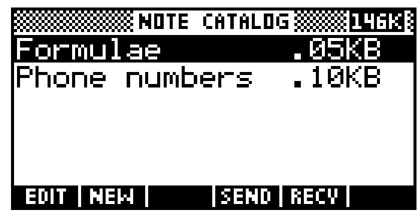
Use the keys discussed above to type in the screen shown on the right. The arrow keys can be used to move around in the text and insert or delete characters.

The **BKSP** and the normal DEL key are used to remove characters. Pressing the DEL key deletes the character at the cursor, whereas **BKSP** removes the character before the cursor.

Generally most characters you will need are on the keyboard, but additional special characters can be obtained through the **CHARS** view. The three pages of the **CHARS** view are shown below. As you can see, there are many special characters available for use.



Once you have finished, and perhaps added some more formulas of your own, just press **SHIFT NOTEPAD** again to return to the catalog level. You will find that your new Note is now listed. There is no need to save your note because it is saved as you work. Just exit when finished. Unfortunately this also means that there is no way to undo a mistake by reverting to a previously saved version.



You may wish to try transferring your Note to a friend's calculator.

- If you own an hp 39gs then use the infra-red link. Press **SEND** and "HP39 (IrDA)" on your machine. Press **RECV** and "HP39 (IrDA)" on your friend's machine.

- If you own an hp 40gs then plug the mini serial cable into both machines. Press **SEND** and "HP39/40 (Ser)" on your machine. Press **RECV** and "HP39/40 (Ser)" on your friend's machine.

The astute reader may have notice the superscripted characters ^{1}, ^{2} and ^{3} in the first screen above. The second is the mathematical operator for squaring. The first is simply a textual character: trying to use it as a mathematical operator will result in "Syntax error". The third is an oddity. If you choose the superscripted 3 then you will find that the character which actually appears is that of the inverse operator $^{-1}$. This character is available on the keyboard above the divide button. The reason for this odd transposition is based in the history of the calculator. The original hp 38g only had the function INVERSE(…) and the $^{-1}$ operator was added for convenience sake in the hp 39g, released in 2000. To do this the creators had to borrow one of the existing unused characters, the ^{3} character, and convert it into the $^{-1}$ operator. However, they forgot to change it in the **CHARS** view and this error has never been fixed in any of the successive models!

An occasional problem users encounter is finding that the Notes they have saved onto a computer are corrupted when they try to read them back. This is almost invariably caused by trying to open, read, print or otherwise tamper with the Note while it is on the computer. If you view the file once it has been stored by double clicking on it to try to open it then the computer may try to edit it with one of the word processors you have installed. If you allow this and save the result then all sorts of characters may be inserted into the file that will make it unreadable by the calculator. If you find that you have unintentionally opened a file in this way then exit from the word processor without saving. More information on transferring to & editing on a PC is given on pages 226 - 252.

WORKING WITH SKETCHES

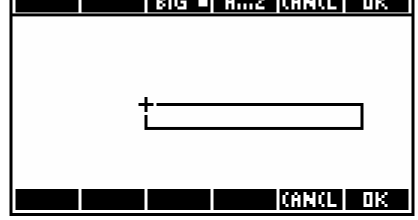
If you have not already done so, read the previous chapter. As is explained there, every aplet has associated with it a Sketch view, made up of a number of pages (the default is one page). It can be viewed by pressing **SKETCH**, located above the **VIEWS** key.

Sketches for the standard aplets start blank but you may find that an aplet that you download from the Internet will have a Sketch attached. Remember a Sketch is always attached to a particular aplet, so changing aplets makes it inaccessible.

Facilities provided in the Sketch view are good for a bit of fun, but very primitive when you try to do anything at all complex. This is not meant as a criticism of the calculator. It does an extremely good job at what it was designed for - working with numbers - but it was never designed to compete with a computer drawing package.

When you first enter the Sketch page on your calculator you will see the view at the top right. There are four screen keys available. The **TEXT** key allows you to place strings of text on the screen.

If you press **TEXT** then you will be prompted to enter a string of text at the bottom of the screen. An **A...Z** key is provided to lock in alphabetic keys.



When you press **ENTER**, a rectangle will appear in the middle of the screen. The rectangle is the same size as the text will be. Using the arrow keys you can now move the text to the position in which you want it and then press **ENTER** again to fix its position.

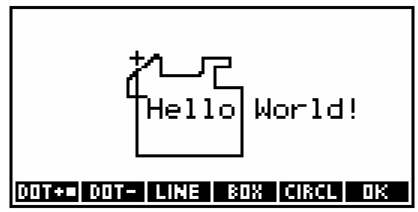
There are two font sizes available via the BIG ■ key, with the default size being large. If you press the BIG ■ key then it will change to BIG. Although there is no apparent change when you are typing in the text, the font will become smaller when it appears in the window. Only uppercase is available in this small font.

New sketch pages can be produced by pressing the NEWP key.

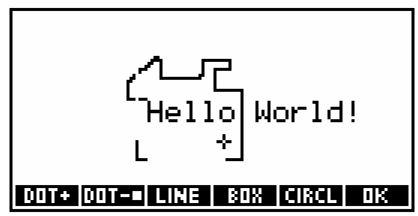
The DRAW menu

The DRAW key gives access to a slightly enlarged menu of simple drawing tools. The OK key seen on the far right exits from this menu back to the original one. The tools provided are very primitive compared to a computer drawing or painting program.

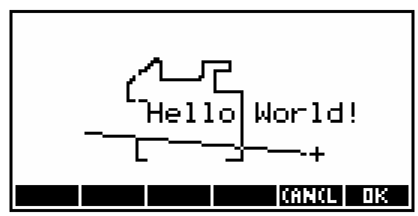
The small cursor (cross) in the middle of the screen can be moved around using the arrow keys. If you press the screen key labeled DOT+ then a trail will be drawn as you move the cursor. Notice the small dot next to the DOT+■ showing that it is engaged.



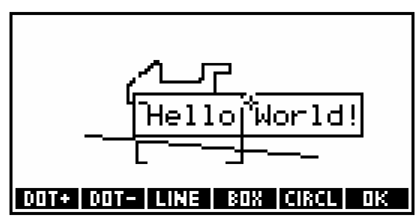
Pressing the DOT- key turns the cursor into an eraser (and automatically disengages DOT+■ as well). Pressing the same key again disengages both keys and leaves the cursor free to move with no effect on the Sketch.



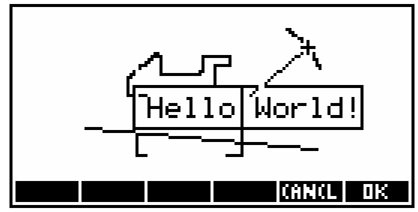
Moving the cursor to one end of a proposed line, you can now press the LINE key and move the cursor to the other end of the line. When the line is correctly positioned, press the OK key (or CANCL).



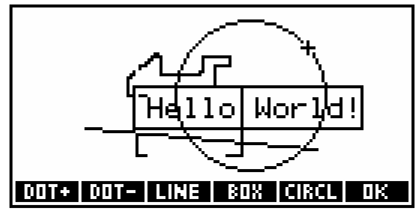
A box is drawn in the same way. Position the cursor at one corner, press the BOX key, move the cursor to the diagonally opposite corner and press OK.



The circle command is similar to the box command. You should position the cursor at the center of the proposed circle. Pressing CIRCL, move the cursor outwards from the center, forming a radius. As you do so you will see a small arc appear, giving you an indication of the curvature of the circle.



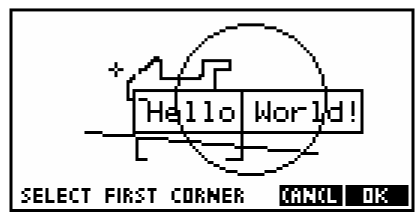
Pressing OK (or **ENTER**) will then complete the circle.



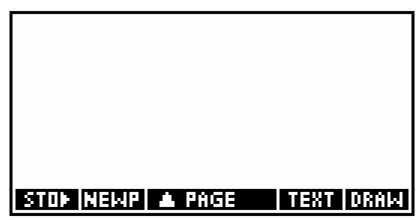
Finally, press OK to leave the drawing tools view.

Using the STO▶ key you can capture part of the screen and store it into any of ten graphics memories **G1,G2..G9,G0** (called 'GROBs', which is short for *graphics objects*). When you press STO▶ the message you see on the right will appear, asking which GROB to use.

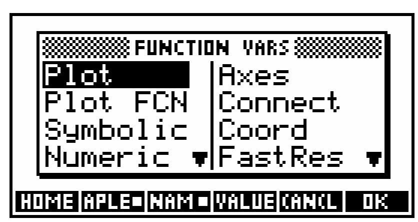
Once you have chosen a GROB in which to store the screen capture, you will need to specify the corners of the rectangle to be captured. Position the cursor on one corner and press OK. Move to the diagonally opposite corner and press OK again.



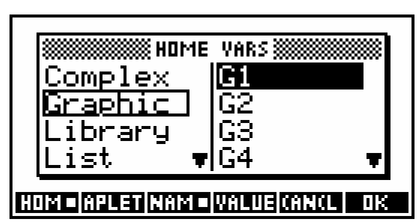
Once an image is captured, it is possible to paste the captured screen portion into the a Sketch page using the **VAR** key. The screen does not have to be blank but to make things clearer, create a new sketch page using the NEWP key (see right). Notice the sudden appearance of a ▲ PAGE key to allow movement between pages.



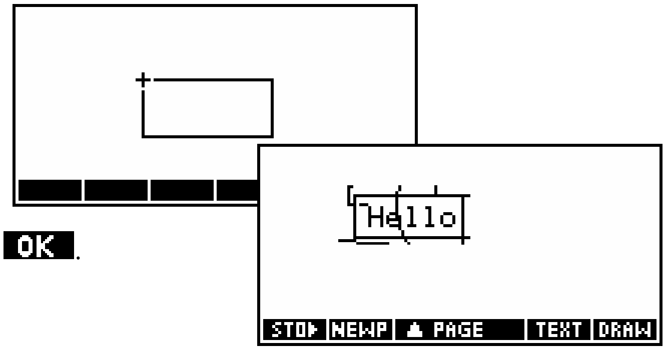
To paste the contents of a stored GROB onto the screen press the **VAR** key and, when you see the screen shown right, press the key labeled HOME. Note that this is not the **HOME** key on the keyboard but the one below the screen.



Move down through the menu until you reach *Graphic* and across to the particular GROB you chose. Now also press the key labeled VALUE and then press OK.



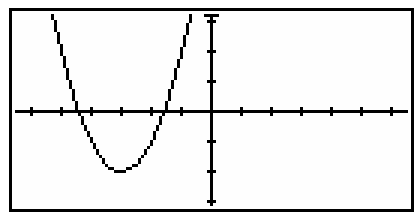
You will now find yourself back in the graphics screen with a rectangle representing the size of the GROB to be pasted.



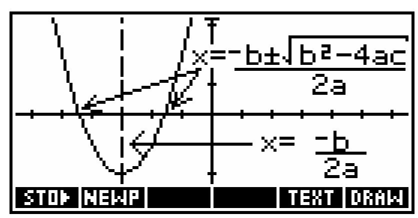
Move the rectangle to the desired position and press **OK**. The contents of the stored GROB will appear.

This pasting technique can be very useful in building a Sketch, particularly when used in conjunction with the ability to capture **PLOT** screens and store them in GROBs (see below). You can also use it to create animations by pasting the same image in successively moved locations on each of the pages. Holding down the **PAGE ▼** key will move rapidly through the sketches and animate them but the speed of the animation may be too fast to work properly. If you can program then you can automate the animation quite simply. An example of a simple programmed animation can be found on the author's website (http://www.hphomeview.com). It consists of a pair of seals tossing a ball back and forth.

To capture a **PLOT** screen, just arrange the **PLOT** display so that it shows the features you wish to capture. For example, if you don't want the screen menu showing, make sure it is set up that way before you proceed. When ready, press *and hold down* the **ON** key, then press the **PLOT** key and release both. The screen will be automatically stored into GROB **G0**. You can then paste it into a sketch.



Change into the sketch view, press **VAR** and select **HOME** rather than **APLET**, from the options at the bottom of the screen. Move the highlight to *Graphic* and then into the right-hand column and find **G0**. Press the **VALUE** key and then press **ENTER**. Unlike the previous example where the pasted GROB now had to be located on the sketch page, the captured screen in this case is a full size image and so will be pasted in as a fresh page rather than replacing part of an existing one.



Having pasted it into the Sketch page, you can now modify it by adding text and other information.

One has to question, however, whether the time needed to do this and the crudity of the result make the whole process worthwhile. If you're intending to do this to produce a set of 'cheat notes' for your next test or exam, you would do better to spend the time studying!

Calculator Tip
The screen capture facility demonstrated here can be used to capture any screen as a GROB, not just a **PLOT** screen. Pressing **ON+PLOT** at any time will store an image of the current screen into **G0**. See page 251.

34

COPYING & CREATING APLETS ON THE CALCULATOR

This chapter assumes a reasonable degree of familiarity with the majority of the built-in aplets.

As has been discussed before, the designers of this calculator provided a set of standard aplets for you to use, changing the capabilities of the calculator as you change aplet. These standard aplets will cover most, if not all, of your requirements but to a certain extent you can also modify them to suit your needs and copy them for your friends. No programming is necessarily required at the lowest level and so, unless you *want* to learn about the programming language of the hp 39gs & hp 40gs, there is no reason to worry about it unless you want to produce highly enhanced aplets.

In this chapter we will cover the creation of aplets and, to a lesser extent, programs and notes on the calculator together with the ability to transfer them to another calculator via cable (hp 40gs) or infra-red (hp 39gs). Generally this will involve making small modifications to the standard aplets and saving them under a new name. In a chapter which follows we will cover the use of software on a PC to do this.

As well as this you can download additional aplets and programs written by people who *do* enjoy programming. These aplets come via the Internet, but you may be able to obtain them from your teacher, from other users, or from a PC onto which they have been copied. Once aplets have been copied from the Internet the USB cable provided with your calculator can be used to download the aplets to the calculator. This will also be covered in a subsequent chapter on page 245.

We will begin by discussing how to send and receive aplets and programs from one calculator to another and then proceed on to how to create them. This may sound backwards but my experience has been that most users arrive at this topic by deciding that they want to know how to send/receive and only afterwards start asking themselves *what* to send/receive!

Different models use different methods to communicate

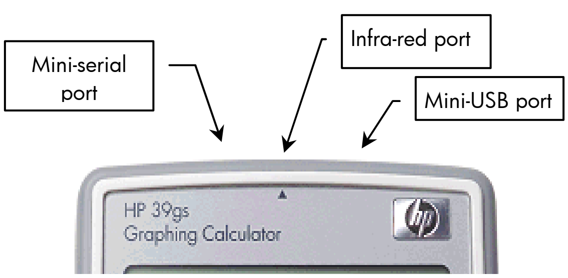
As has been discussed on page 7, the hp 39gs and hp 40gs were aimed at different markets. Both of them require communication with a PC and so <u>both</u> come with a standard "mini-USB" port and cable. The mini-USB cable (shown left) is identical to one commonly used with digital cameras and replacements can be bought very cheaply in any electronics store.

Mini USB cable

Mini Serial cable

Although communication with a PC is identical on both models, the similarity ends there. On the hp 39gs communication between calculators is accomplished using the infra-red port. This is very similar in operation to that of a remote control. Communication is only possible over a very short distance – up to about 8cm.

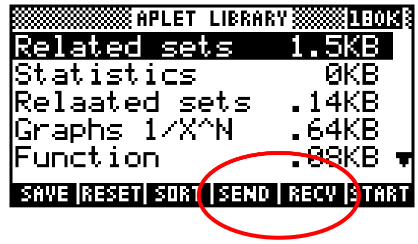
In the markets for which the hp 40gs was designed infra-red communication was not considered desirable and a mini-serial cable connection is used instead. This cable (shown above right) connects two hp 40gs calculators via their mini-serial ports. It is an HP proprietary cable which can only be purchased from HP or an HP reseller. Both cables are supplied with the hp 40gs calculator as standard equipment but the hp 39gs is supplied standard with only the mini-USB cable because there is normally no need for the mini-serial cable with an hp 39gs. The mini-serial port is still present though because it is anticipated that it may be used in the future to connect to data-loggers and other possible peripherals.



If you are using the cable on an hp 40gs then alignment of the calculators is obviously not a problem. For infra-red, of course, it is. Look at the keyboard side of the calculator, near the HP label above the screen, you'll find a small black triangle. This marks the position of the infra-red port so that you can line up calculators when looking down from above.

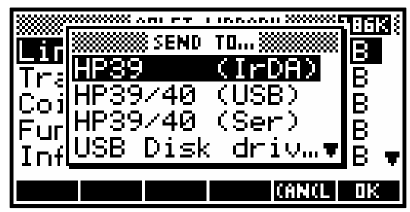
Sending/Receiving via the infra-red link or cable.

Any aplet, note, program, matrix or list can be copied from one calculator to another via the infra-red link at the top of the calculator on an hp 39gs or via the supplied cable on an hp 40gs. A sketch can be transferred by sending the aplet to which it belongs.

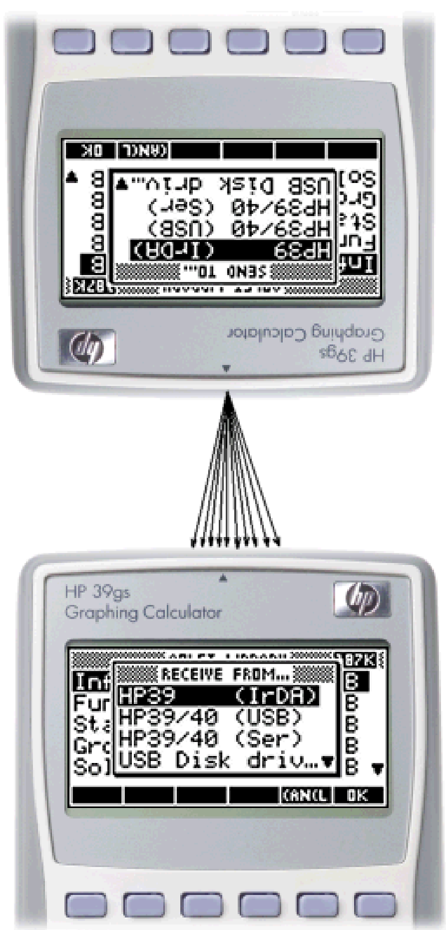


The key to this ability is the screen key labeled **SEND** and its companion key **RECV**. This is shown in the **APLET** view on the right. These keys can be used to send a copy of the highlighted aplet to any other calculator. These **SEND** and **RECV** keys also appear in other views.

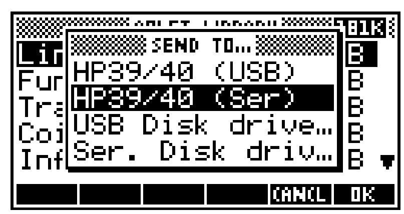
If using the infra-red on the hp 39gs then you should begin by putting the two calculators top to top with the calculators no more than 8 -10 cm apart and line up the arrows. The best separation is about 3 - 4cm. It is best not to put them extremely close together or touching because in that position a small misalignment will cause most of the signal to be lost. Placing them 3 - 4cm apart allows the beam to spread and compensates for any slight misalignment.



When you press **SEND** you will see the menu shown right. The option you need for an hp 39gs is "HP39 (IrDA)", which transmits via the infra-red link. On the hp 40gs this option is missing.



If using the supplied mini-serial cable to copy from one hp 40gs to another then you simply need to plug the cable into the mini-serial ports at the top of each calculator. Make sure you are using the correct cable and the correct port (see the previous page).



When you press **SEND** you will see the menu shown right. The option you need for an hp 40gs is "HP39/40 (Ser)", which transmits via the mini-serial cable.

The diagram on the left is, of course, for a pair of hp 39gs calculators using infra-red. To send an aplet, both calculators should be showing the **APLET** view, *with the highlight on the aplet you wish to send.*

The process is essentially:

- Press the **SEND** key on the sending calculator and the **RECV** key on the receiving calculator.

- Choose the option for your particular calculator. On an hp 39gs this will be "HP39 (IrDA)" and on an hp 40gs it will be "HP39/40 (Ser)".

- Press **OK** on both calculators and the transmission process will begin.

Calculator Tip

If you begin to receive "Time Out" messages when transmitting aplets or notes then this means that the connection was lost and the transmission failed.

The two most common causes of this are:

- mis-alignment of the two calculators. The infra-red beams must be correctly lined up and within 8-10 cm.
- low batteries. In my experience the transmission process begins to act up a little before the low battery indicator icon appears at the top of the screen.

When replacing the batteries remember that batteries that have been sitting on the shelf in a supermarket for 12 months or more may be nearly flat despite never having been used!

Creating a copy of a Standard aplet.

Imagine either of these two scenarios….

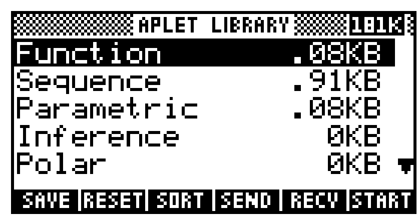
- you are a student and you have filled the Function aplet with a set of equations needed for tonight's homework and set up the **PLOT** screen so that it looks exactly the way you want it to. Now you find that you need the Function aplet to do something else equally important which will mean wiping all that work.

- you are a teacher and you are planning a lesson where you will examine a collection of about half a dozen data sets and graph the results. You don't want to spend half the lesson waiting while the students type the data into the Statistics aplet and then watch while they all use different axes and get totally different graphs.

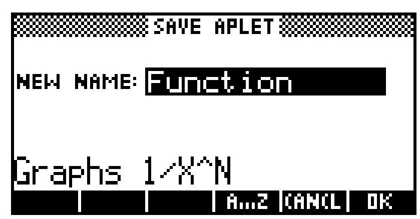
In either of those two cases, the solution is to make a *copy* of the aplet concerned. You can make as many copies of any of the standard aplets as you wish. The only limit is the calculator's memory. Depending on what you put in them the calculator's memory is normally sufficient to store anything from 30 to 100 aplets.

Let's look at each of the two scenarios in turn.

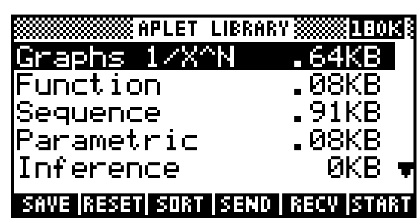
In the first case, what the student needs to do is to make a copy of the Function aplet to hold his homework (the functions he had already set up) and then do the unexpected extra work in the original Function aplet which is now free.



Press the **APLET** key to see the list of aplets. Move the highlight to Function (or whichever one you wish to copy) and press the screen key labeled **SAVE**.

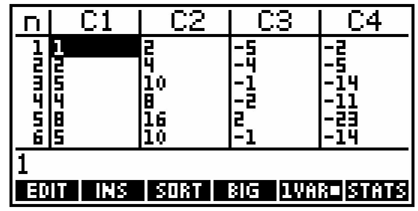


You will now be asked to nominate a name for the newly created aplet. It is a good idea to name it something that will remind you of its purpose and contents later. After all, you may end up saving it permanently onto a computer using the Connectivity Kit. When you look at it six months from now a name of "Homework" is not going to tell you much.

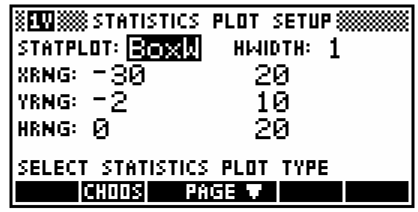


Your name can contain spaces and any other characters, including those from the **CHARS** menu. Names can be of any length but the aplet library view only displays the first 14 characters.

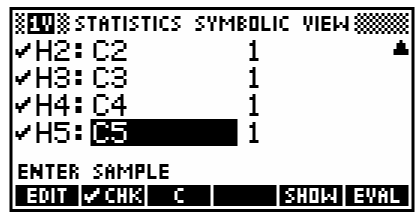
Our student's newly created copy of the Function aplet is now totally independent of its parent aplet. The student can now (if desirable) **RESET** the original Function aplet back to factory defaults and go on with the extra work that she wanted to do. A saved aplet cannot be **RESET** because the calculator is no longer sure what to reset it to. To remove a saved aplet you need only press DEL. Pressing DEL on a normal aplet is equivalent to **RESET**.



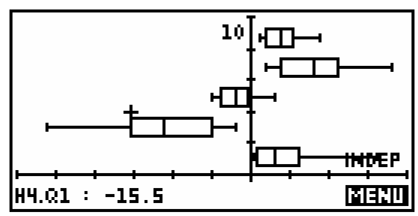
Our second scenario had a teacher not wanting to waste the time that would be needed for his students to type in five sets of data. In this case also, the solution is to make a copy of the aplet.



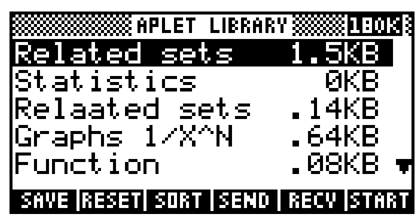
First set up the Statistics aplet to be exactly the way you want it… for example five data sets of 20 numbers, **NUM** view set to univariate stats (**1VAR**), **SYMB** view set up for five *Box and Whisker* graphs, axes set up to display all five sets. See page 114 for information on the Statistics aplet.



The next stage is to move to the **APLET** view and **SAVE** this modified Statistics aplet under a new name - say "Related Sets". If desired, you can **RESET** the original Statistics aplet afterwards, ready for next use.



This saved aplet can now be downloaded to all the students' calculators using the infra-red link in the case of an hp 39gs or using the cable in the case of the hp 40gs. This ensures that each student has exactly the required data sets, with **PLOT** views pre-set to the teacher's needs.



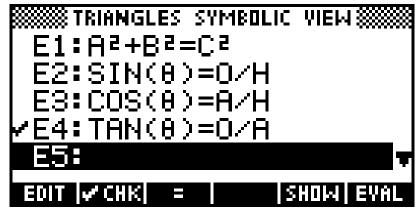
An ambitious teacher might even complete his aplet by adding a set of instructions to the Note which is associated with the aplet. This would allow an absent student to download a copy from a friend's calculator and then find instructions on what they are supposed to do in the Note.

Indeed, if the students have access to the Internet and a Connectivity Kit themselves, then there is no reason that the teacher could not post the aplet on the department's web page for downloading by any students who need access due to absence from class.

In both of these cases, the procedure has been to save a copy of one of the standard aplets under a different name. In neither case was there any need to do any programming and the amount of memory taken up by these copies is fairly minimal because they share most of their resources with their parent aplets.

Some examples of saved aplets

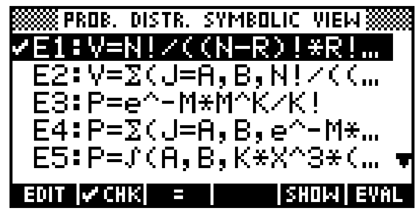
The Triangles aplet



In the **APLET** view, **RESET** the Solve aplet and **SAVE** it under the new name of “Triangles”. Now **START** it and enter the formulas shown.

The theta character can be obtained from the keyboard on the zero button using **ALPHA**. Change into the **MODES** view and set the angle mode to *Degrees* (unless you want to use another mode). By changing into the **NUM** view you can now use this to solve problems in right triangles. Some users choose to use the letter **D** for degrees rather than theta because theta can be mistaken for an **O**.

The Prob. Distributions aplet



In the **APLET** view, **RESET** the Solve aplet and **SAVE** it under the new name of “Prob. Distr.”.

This aplet is for use by students studying probability distribution functions, and contains formulas that can be used to perform calculations on the Binomial, Poisson, Exponential and Normal distributions. You can adapt it to the requirements of your particular course.

Firstly, **START** the aplet and enter the formulas shown below.

E1: **V=N!/((N-R)!*R!)*P^R*(1-P)^(N-R)**
E2: **V=Σ (J=A,B, N!/((N-J)!*J!)*P^J*(1-P)^(N-J))**
E3: **P=e^-M*M^K/K!**
E4: **P=Σ (J=A,B,e^-M*M^J/J!)**
E5: **P=∫(A,B,K*X^3*(2-X),X)**
E6: **P=e^-(K*A)-e^(-K*B)**
E7: **P=UTPN(M, S^2,X)**
E8: **P=1-UTPN(M,S^2,X)**
E9: **P=UTPN(M,S^2,A)-UTPN(M, S^2,B)**
E0: **P=UTPN(M,S^2,M-K)+ UTPN(M,S^2,M+K)**

These formulas can be used in the **NUM** view to solve problems involving the probability distributions listed earlier. Some explanations and examples are given on the next page.

Note that equations **E1** and **E2** use the expression **N!/((N-R)!*R!)** instead of **COMB(N,R)** because doing this allows backward solving to solve for **N**. Using **COMB(N,R)** only allows the formula to be used in a forward direction. The drawback of this is that values of **N** above roughly 75 will cause overflow internally and result in errors. However, values like this should probably be done using a Normal approximation anyway. If this is a problem, replace the **COMB(N,R)**. Additionally, the letter **V** for value is used in these two equations because, for this particular distribution, the variable **P** has the specific meaning of “the probability of individual success”.

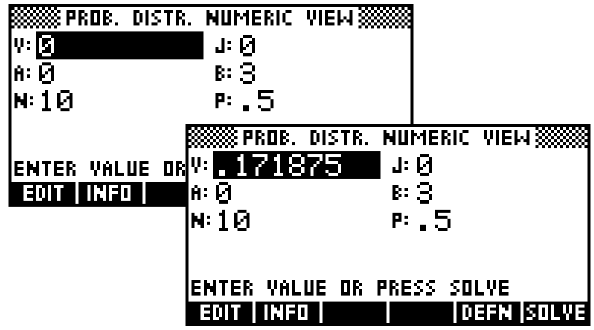
Equations **E1** and **E2**

These two equations can be used for calculations involving individual and cumulative Binomial probabilities.

eg. Find the probability of at most 3 heads when tossing a coin 10 times.

Ensure that formula **E2** is **✔CHK**ed and enter the values shown right. The value in **J** is irrelevant as it is merely the summation variable. Highlight **V** and press **SOLVE**.

Answer: 0.1719



Note: If N is larger than 200 then you should use a Normal approximation. The **N!/((N-R)!*R!)** section of the formula will cause internal overflow and inaccurate answers above this level. The time needed to perform the summation will probably also be excessive. Solve works by repeated iterations converging on the correct value and this will be quite slow if the summation has many terms. If it is important not to use a Normal approximation then replace the **N!/((N-R)!*R!)** portion with **COMB(N,R)**. The **COMB** function has special facilities for handling large numbers. This will not help greatly with the iteration time needed.

Equations **E3** and **E4**

These two equations can be used for calculations involving individual and cumulative Poisson probabilities, where **M** is the mean. The technique is otherwise identical to the Binomial problem.

Equation **E5**

The formula in **E5** is a generic formula for problems such as the one below:

"A probability distribution has the equation $f(x) = 0.625x^3(2-x)$; $0 \le x \le 2$. Show that this is a valid probability distribution function and use it to find $P(x \le 1.2)$"

Use it by substituting whatever function is in use for the one currently entered. As this formula involves the integration function, each use of the solve process will require the calculator to perform multiple integrations. Because of this the solving process will be relatively slow.

Equation **E6**

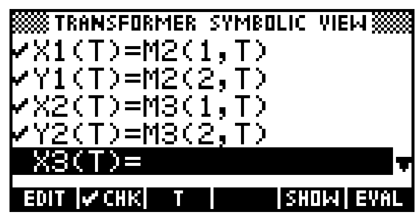
This equation gives $P(a \le x \le b)$ for an exponential distribution. To calculate $P(x \le a)$ use $P(0 \le x \le a)$. To calculate $P(x \ge a)$ just find $P(x \le a)$ and then use the **HOME** view to calculate the complement.

Equations **E7** and **E8**

Finally, equations **E7** to **E0** concern the Normal distribution, with **E7** giving $P(X \ge x)$, **E8** giving $P(X \le x)$, **E9** giving $P(a \le x \le b)$ and **E0** allowing calculation of questions such as "what distance either side of the mean will give a probability of 0.45?".

Finally, you may choose to split this aplet into two, placing equations 1 to 4 into an aplet called "**Discrete PDFs**" and the others into another called "**Cont. PDFs**". I encourage my students to do this because it reinforces the correct technique of first of identifying whether the problem is discrete or continuous.

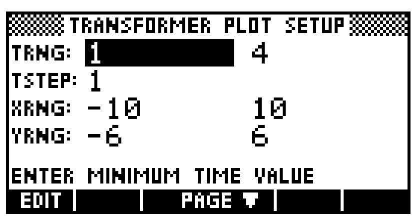
The Transformer aplet

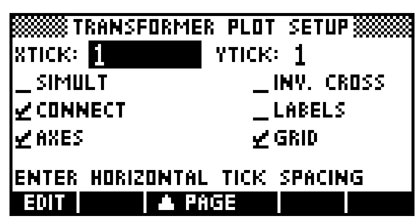


This aplet is based on the Parametric aplet and allows students to investigate geometric transformations using 2x2 matrices.

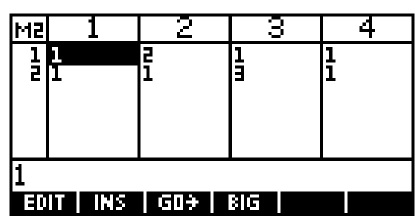
In the **APLET** view, **RESET** the Parametric aplet and **SAVE** it under the new name of "Transformer". Enter the equations shown right.

Change to the **PLOT SETUP** view and enter the settings shown.



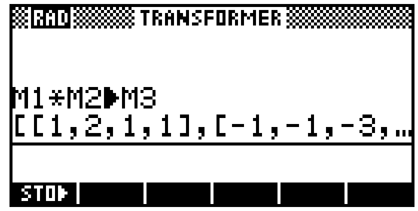


Now change to the **Matrix Catalog** view and enter the matrices shown below into **M1** and **M2**.

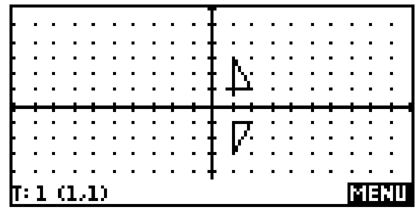


$$M1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad M2 = \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & 1 & 3 & 1 \end{bmatrix}$$

Change to the **HOME** view and perform the calculation shown right and finally press **PLOT**.

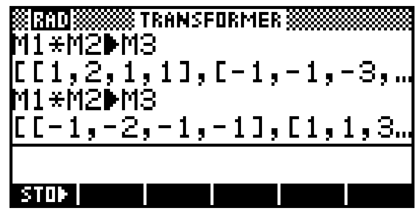


The result is a triangle with corners at (1,1), (2,1) and (1,3), along with its image after reflection in the x axis.

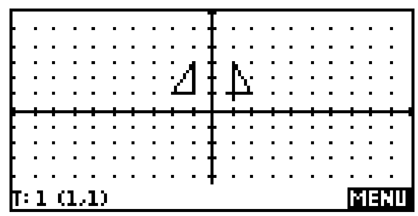


We can now **EDIT** matrix **M1** so that it contains another matrix.

For example: $\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$



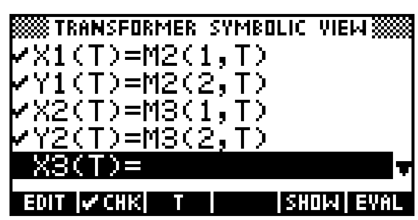
To see the effect of this new matrix, simply return to the **HOME** view, **COPY** the previous calculation and press **ENTER**. The new image will be stored into matrix **M3**. If you now return to the **PLOT** view the image will not appear to have changed as the aplet does not realize the matrix has changed but pressing **PLOT** again will force a re-draw of the new image.



The great power of this aplet is its use as a teaching and investigative tool. Simply continue to **EDIT** matrix **M1**, repeating the **HOME** calculation, and re-**PLOT**ing each time to see the change in the image.

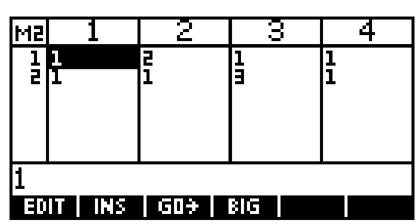
For teachers, the degree of guidance which should be given to a class will obviously depend on their level of ability. For an able class you might choose to give no more guidance than to suggest that they confine their investigations initially to placing numbers only on the diagonals. It may be a good idea to challenge them to record their matrices on the board as they discovered them. For a less able class you might hand out a list of matrices they should investigate. Perhaps in the form of a set of cards with matrices and geometric transformations which must be matched up. A highly able class will find nearly all relevant matrices within 20 to 30 minutes. A less able class may need considerable guidance.

So… how does this aplet work?



The formulas in the **SYMB** view form the key to the process by allowing the calculator to fetch values from the matrices, with the values fetched being determined by the settings in the **PLOT SETUP** view.

For example, as **T** runs from **1** to **4** in steps of **1** the **(X1,Y1)** values plotted become (M2(1,1),M2(2,1)), (M2(1,2),M2(2,2)), (M2(1,3),M2(2,3)) and (M2(1,4),M2(2,4)). If we now substitute the actual values from matrix **M2** then these points become (1,1), (2,1), (3,1) and (1,1), which give the shape when plotted.



The repetition of the first point is to ensure that the line forming the triangle is closed by connecting back to its starting point. The function formed by **X2(T)** and **Y2(T)** perform the same function with matrix **M3**.

To use a different shape you need only change the points in matrix **M2**. If your new shape has more than three vertices you will need to change the TRange values in the **PLOT SETUP** view. It is not a good idea to use a shapes that is highly symmetrical, like a square, as it makes it harder to recognize transformations.

Calculator Tip
It is probably faster to have the class set it up themselves instead of sending it via cable or infra-red. Plus, if the class is not to regard this as 'magic' then it helps if they understand how the aplet works. This is best accomplished by having them set it up themselves. You might even consider allowing them to first play with drawing shapes of their own invention using a single matrix and only X1 and Y1. This will show clearly why the TRange settings are important, and how the values are extracted from the matrix.

35

STORING APLETS & NOTES TO THE PC

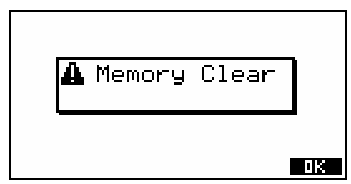
Overview

Many users create elaborate collections of notes and aplets over time, particularly if they begin to share resources with their friends. Students may load aplets onto their calculator at the request of a teacher and wish to keep them to refer to later. As time goes by the calculator can become increasingly cluttered and the amount of memory (234Kb) is such that it will hold a lot of information!

In view of this it is quite a good idea to save your notes and aplets onto a PC using the supplied cable and software. By doing this you free up space on your calculator and, more importantly, you ensure that the material is backed up onto a PC and is safe from accidental loss.

In addition to this, one of the nice features of the hp 39gs and hp 40gs is that when you switch it on it resumes operation in exactly the state you left it. Not all calculators will do this and it is clearly highly desirable but it also has its downside. Effectively it is somewhat similar to always placing your PC into "Standby" mode rather than actually shutting it down. The advantage is that it resumes exactly as it was when you turned it off but you would normally not do this on a PC day after day, month after month without expecting to experience problems. The PC's operating system may be quite stable but any system that complex needs to be completely shut down and restarted at least once a week or once a fortnight.

The same is true of the hp 39gs and hp 40gs. Its operating system is, of course, not as complex as that of a PC but despite this if you continue to use it without shutting it down and restarting it periodically then it may eventually become unstable after a long period. It should be stressed that this may not happen at all and even if it does it will take many months or even years to occur.



If there is a problem it can appear in a number of ways. Some of the common symptoms are that the calculator will freeze (lock up) or, worse, that it will spontaneously reset itself, clearing its memory in the process. Having this happen to you in the middle of an examination is obviously not desirable!

Fortunately the solution is fairly simple. Just as you should periodically back up your PC you should also do the same on your calculator. Save any aplets, notes, lists or matrices that you want to keep to the PC and then perform a hard reset (memory clear) to freshly reload the operating system from the chip. Then restore any of the information that you need from the PC. See page 42 for instructions on how to perform a hard reset.

The pages which follow give instructions on how to use the supplied software to save material from the calculator to the PC and back.

Software is required to link to a PC

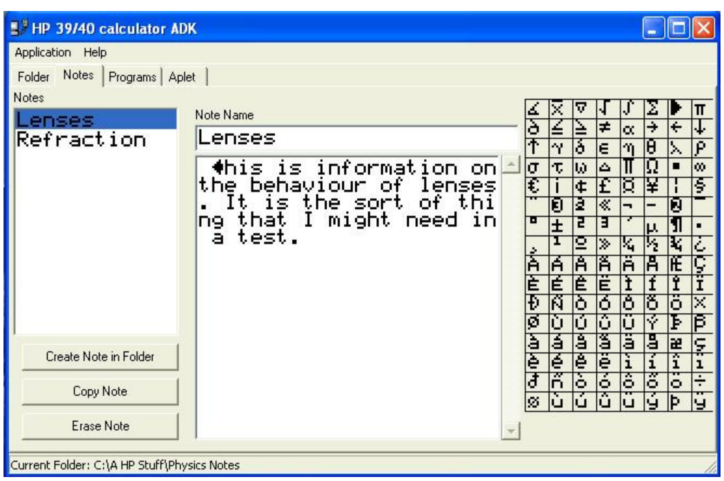
The connectivity software for the hp 39gs and hp 40gs was being rewritten at the time when this book was being published. The version on the CD which came with your calculator may not be the most recent version. For the latest version of the software for your calculator you should consult Hewlett Packard's web site (http://www.hp.com/calculators) or the author's website at http://www.hphomeview.com (this site tends to be updated more often). Unfortunately this software is only available for Windows.

For the hp 38g, hp 39g & hp 40g

The three older models listed above all use a serial port to communicate with the calculator. This method was very stable and reliable but since about 2003 it has become increasingly common for computers, particularly laptops, to be sold without a serial port. Instead the USB port has become the standard means of communicating with peripherals such as printers, scanners, cameras and calculators. All recent HP calculators have used a USB connection.

The HP Connectivity Kit, called "HPGComm", is the software used for the older serial port models. Because it is only of use to these earlier models it will not be discussed here. Information on using this software can be found in a variety of places, such as the author's website at http://www.hphomeview.com. Although it will transfer information, notes, aplets and programs to and from the calculator it does not let you edit them in any way. To do that you need to use a program called the ADK. This can also be found at the author's website together with instructions.

For the hp 39g+, hp 39gs and hp 40gs



As stated earlier, at the time of writing in mid-2006 the hp 39gs and hp 40gs had only just been released and new software was in the process of being developed. The version finally released may differ in small ways from that shown on the pages following but will be substantially as shown. Both models are supplied with a mini-USB cable that lets them link to the USB port of a PC. The software allows the transfer of objects such as aplets, programs or notes and also allows you to edit them once they have been transferred to the PC.

Always check to see if there is a newer version of this software available.

Both models use the same cable

As has been discussed elsewhere the hp 39gs and hp 40gs were aimed at different markets. Both of them require communication with a PC and so both come with a standard "mini-USB" port and cable. The mini-USB cable, shown left, is identical to the one commonly used with digital cameras and replacements can be bought very cheaply in any electronics store.

This mini-USB cable is used for communication with a PC and the same cable is used for both the hp 39gs and the hp 40gs via the USB cable on the PC.

Be careful when plugging the cable into the calculator that you are using the correct port. The port arrangement for an hp 40gs is shown right. An hp 39gs is identical except that the space between the two ports contains the electronics for use in infra-red communications.

Sending from calculator to PC

Any aplet, note, program, matrix or list can be copied from a calculator to a PC using the supplied mini-USB cable. A sketch can be transferred by copying the aplet to which it belongs.

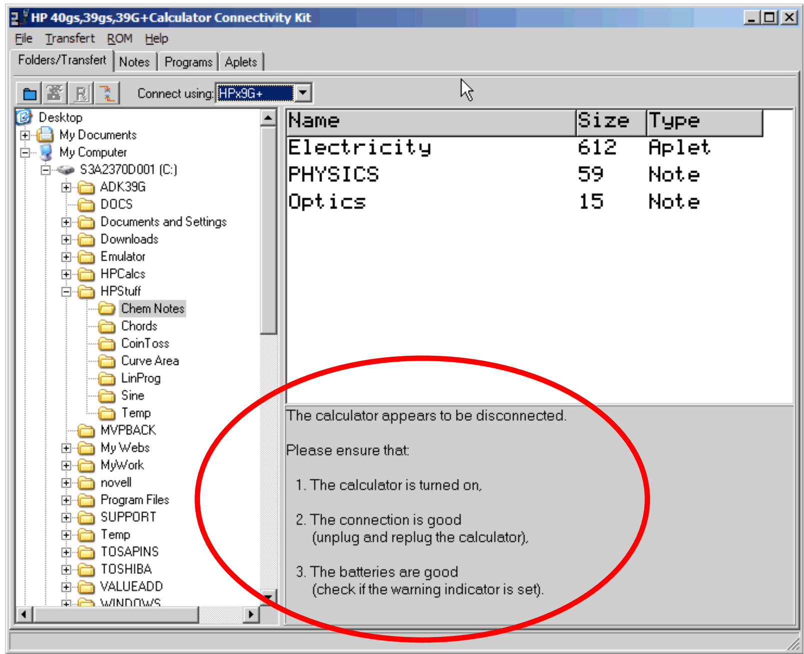
The key to this ability is the screen key labeled **SEND** and its companion key **RECV**. This is shown in the **APLET** view on the right. These keys can be used to send a copy of the highlighted aplet to the PC. These keys also appear in other views.

The process is essentially:

1. Highlight the aplet (note, list, program etc) on the calculator that you want to send to your computer.
2. Run the software on the computer and select the folder in which to store your aplet.
3. Press the **SEND** button and highlight the "USB Disk drive…" option from the pop-up menu.
4. Send the aplet by pressing **ENTER** on the calculator.

This is covered in more detail on the pages following.

Before beginning you should install the Connectivity software. This can be found on the CD that came with your calculator but it is best to download a fresh version from the web so as to obtain the most recent version (see page 237). Begin by plugging the cable into the mini-USB ports at the top of the calculator. The other end plugs into any vacant USB port on the PC. Make sure you are using the correct cable and the correct port (see the previous page).



As you can see in the screen capture to the right, before you connect the calculator a message is displayed to indicate that there is no connection.

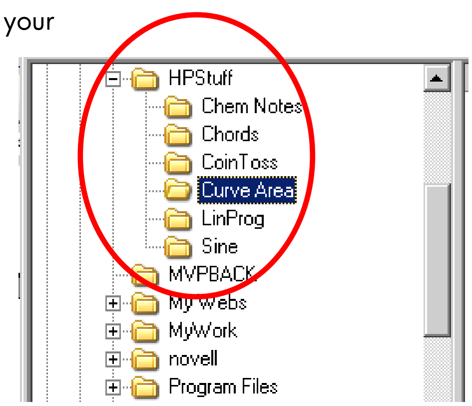
When you plug the calculator in you should find that this message disappears. If it doesn't then there are a number of possibilities.

i) Your calculator may not be turned on. There's an obvious solution to this problem!

ii) The connection may not have properly been established. Try turning the calculator off and then on. Try closing the program and re-starting it with the calculator already turned on. Sometimes it can take up to 10 seconds for the connection to stabilize.

iii) The batteries may not be sufficiently fresh. Communication with the PC uses a lot of power and problems may occur with low battery power even before the Low Battery Indicator appears above the screen on the calculator. You should also be aware that even supposedly new batteries can be faulty if they have been sitting on the supermarket shelf for years before you bought them! Always purchase the best quality.

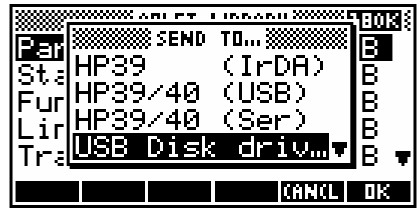
You now need to make a choice which folder you want to use to contain your saved your aplet (or note or…). This is done by simply clicking on the folder you want to use.



It is a very good idea to set up some kind of logical structure to hold your aplets and notes. It is possible to simply dump them all in one folder but this will cause problems if the number of items rises above 40 or 50 simply because it increases the time involved in the communications process (the PC sends a complete list to the calculator each time you **RECV**).

A better idea is to create separate folders for different topics as shown here.

The next stage is to use the software to transmit the aplet, list, matrix or note to the PC. The instructions which follow apply to the transmission of an aplet via the **APLET LIBRARY** view but the process is the same for any other view which has **SEND** and **RECV** keys such as the List view, Matrix view, Program view and Notepad view. To receive or send an object you must be in the appropriate view.



Ensure that the calculator is in the **APLET** view, with the highlight on the aplet you wish to send to the PC. This can be one of the standard ones or one that you have saved under another name.

Press the **SEND** button on the calculator to see the view shown right, which lists the choices on an hp 39gs. On an hp 40gs the "**HP39+ (IrDA)**" will be missing because the hp 40gs has no infra-red ability.

Ensure that the "**USB Disk drive…**" option is highlighted as shown. This is the option which is used for communication with a PC.

The choices on this menu are:

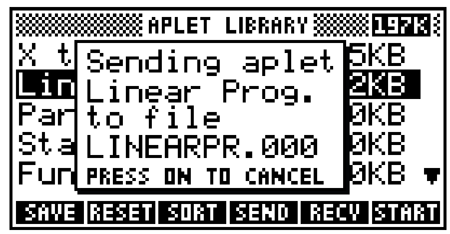
- **HP39 (IrDA)** - infra-red communication from hp 39gs to hp 39gs. *This option only appears on the menu for an hp 39gs.*
- **HP39/40 (USB)** - not used at present. This option would allow cable communication from calculator to calculator via a double ended mini-USB cable. However no such cable is currently part of the equipment supplied.
- **HP39/40 (Ser)** - Used for cable communication from calculator to calculator on an hp 40gs using the mini-serial cable supplied with that model.
- **USB Disk drive…** - Used for communication from calculator to PC using the mini-USB cable. This is the option you will most often be using.
- **Ser. Disk drive…** - This option will probably be used at a later time for communication with external devices such as a data-logger but is currently unused.

When you have chose the "**USB Disk drive…**" option you should then press **ENTER**. The calculator will respond with the menu shown right.

Ignore the "**Pick location…**" option and simply select "**Send now**" by pressing **ENTER** again. Never use the second option. It is simply not worth the space needed here to explain this superseded option and it should be ignored.

Normally the result of this will be a series of small pop up boxes on the PC showing the progress of the file transfer. Since most objects on the calculator are small the pop ups don't tend to stay visible very long. Some aplets will take up to 45 seconds to transmit if they contain large amounts of data or if they contain linked programs. Don't unplug until you are SURE that everything has been transmitted. Sometimes there is an extended pause between files.

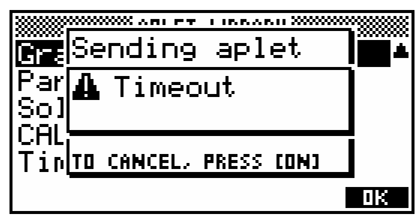
If this is the first time that you have used the folder you chose then you will see the screen shown right. Two small files are normally created in the folder to hold a record of what has been stored there. This screen is simply asking if it is alright to create these files. The files are called HP39DIR.CUR and HP39DIR.000. They should not be edited or interfered with in any way or the record contained in them may be damaged or lost.



During the file transfer process the calculator will also show various pop up boxes to show progress. Their contents depends on what is being transmitted.

The filename that is used on the PC to store the object is normally contracted as shown in the screen to the right. This is not important since, thanks to those "two small files" mentioned in the previous paragraph, the full name will be recorded and restored when transferred back to the calculator. The reason for the contraction is that the original hp 38g from which the hp 39gs and hp 40gs are derived was released for an earlier version of Windows that did not allow long filenames. This has never been changed on later models.

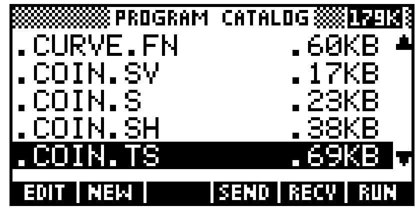
Time out



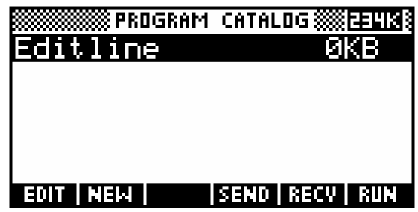
If you see a message saying that there has been a "Time-out", it means that communication with the PC has been lost. This can be due to low batteries. Communication with the PC is very power intensive and problems may occur before the low battery indicator lights up at the top of the calculator screen. It can also be due to problems with the communications software. The USB connection seems to be quite unstable and you may find that you need to close the program and re-start it at times to restore communications.

Attached programs

If your aplet is one that has been given to you by someone else such as your teacher, rather than simply a copy of one of the standard ones, then it may have one or more ‘helper’ programs associated with it.



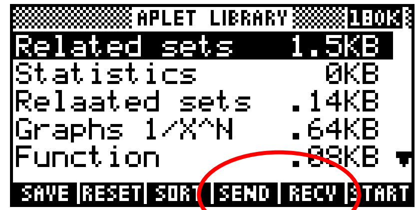
For example, almost all the aplets available from the Hewlett-Packard web site come with sets of up to 6 or 7 programs to do the work, and without which they are totally useless. The screen shown right contains a number of programs which belong to an aplet called “Coin Tossing” which can be downloaded from the web site *The HP HOME view* (at http://www.hphomeview.com).



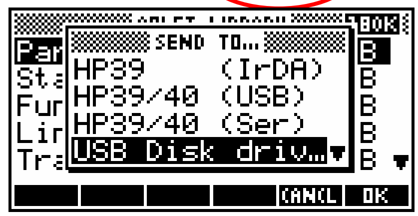
Normally you do not need to worry about this, since the calculator knows they belong with the aplet and will automatically transmit them with it. This can greatly increase the transmission time and it is important that you don’t interrupt the process early. If you want to see these ‘helper’ programs, press **SHIFT PROGRAM** to see a list of the programs currently on your calculator. Even if there are no other programs currently stored, you will always see the ‘Editline’ entry. It contains a record of the last calculation you did in the **HOME** view and can’t be deleted.

Apart from curiosity, there is one important respect in which you need to know about these programs, and that is when it comes time to delete an aplet. The helper programs must be deleted from the **Program Catalog** manually after deleting the main aplet in the **APLET** view. For more information on this see page 250.

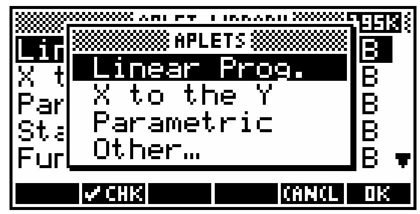
Receiving from PC to calculator



The process of retrieving objects that have been stored to the PC is almost identical to that of sending them in the first place.



Connect the calculator, run the software and choose the folder in which the aplets, notes, or other objects are stored. Then press the **RECV** button and again choose the "**USB Disk drive…**" option from the menu.



The Connectivity software will respond with a list of objects contained in the folder you selected.

If you only want to download one of them then just highlight it and press **ENTER**. To download multiple objects use the **✓CHK** button to select them.

> **Calculator Tip**
> The displayed list will contain only objects that are appropriate for the current view of the calculator. The folder may contain many aplets or programs but if you press **RECV** in the Matrix view then you will see only matrix objects or perhaps nothing at all.

APLETS FROM THE INTERNET

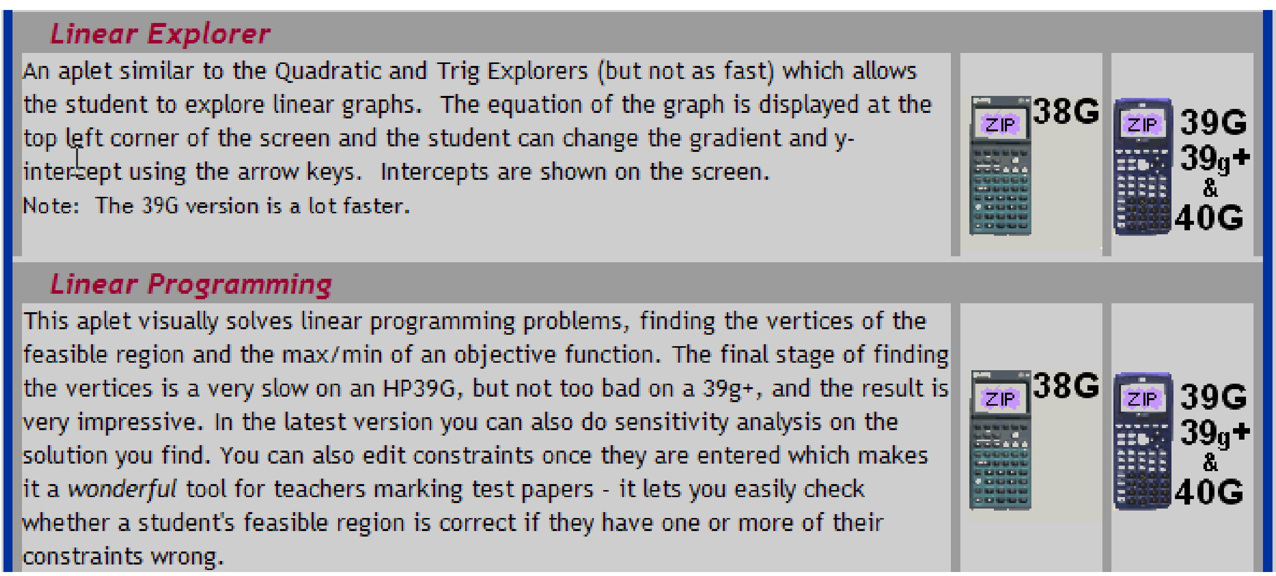
The calculator comes with a number of aplets built into the chip. In addition to these there are hundreds of aplets available to do things such as explore graphs, solve vector problems, explore matrices or analyze time series data, as well as many common tasks called for in Physics and Chemistry. Some of these aplets are straight forward and task oriented. Others are designed to be teaching aplets which allow you to explore concepts and learn for yourself.

The Quadratic and Trig Explorer aplets, now built into the chip, were once teaching aplets which had to be downloaded from the web. Each successive model adds another one or two new aplets to those which are 'standard'.

Finding aplets

The Hewlett-Packard site is one possible starting point and can be found at http://www.hp.com/calculators. From that point you can follow the links to collections of material for the hp 39gs and/or hp 40gs as well as to software and utilities. Other sites can be found that have been created by enthusiasts. One of the most extensive is that of the author, *The HP HOME view*, found at http://www.hphomeview.com.

A small excerpt from *The HP HOME view* is shown below.



Linear Explorer

An aplet similar to the Quadratic and Trig Explorers (but not as fast) which allows the student to explore linear graphs. The equation of the graph is displayed at the top left corner of the screen and the student can change the gradient and y-intercept using the arrow keys. Intercepts are shown on the screen.

Note: The 39G version is a lot faster.

ZIP 38G | ZIP 39G 39g+ & 40G

Linear Programming

This aplet visually solves linear programming problems, finding the vertices of the feasible region and the max/min of an objective function. The final stage of finding the vertices is a very slow on an HP39G, but not too bad on a 39g+, and the result is very impressive. In the latest version you can also do sensitivity analysis on the solution you find. You can also edit constraints once they are entered which makes it a *wonderful* tool for teachers marking test papers - it lets you easily check whether a student's feasible region is correct if they have one or more of their constraints wrong.

ZIP 38G | ZIP 39G 39g+ & 40G

You may notice separate download icons for the 38G and for the 39G, 40G and 39g+ with no mention of the new hp 39gs and hp 40gs. This will change as the sites update the contents to reflect the new models.

In general, any aplet which is suitable for the older HP39G, HP40G or hp 39g+ will also work on the new hp 39gs and hp 40gs. Some games may not be for two reasons. Firstly the earlier models used a slower chip and this means the older games sometimes run so fast that they are unplayable. Secondly, some of them directly access the calculator's chip. If the address on the chip they reference has changed from the old model to the newer one then running the aplet may cause the calculator to lock up or spontaneously reset. The worst result will be loss of user memory. None of the aplets designed for the earliest model, the HP38G, will work on older models.

If you own a calculator then you will already have the required cable with which to download from the internet. If you bought yours second hand without a cable then you'll need to purchase a cable from an electronics store. The mini-USB cable required is the same as that used by many digital cameras.

Downloading an aplet from the web is very simple. Any site will present you with a page which may be similar to the one on the previous page. It may contain either programs or aplets or both. Generally you will be able to click on a link that lets you download that aplet as a compressed ZIP file.

A ZIP file is a special type of file which contains one or more files in compressed. The reason for the compression is simply to allow you to download them from the Internet as one single file instead of having to download each one of the collection separately. The ability to expand these ZIP files is built into Windows XP and you should de-compress them as soon as you have them on your PC. Just double click on the file and it will open as if it is a folder. You should then move or copy them into a normal folder (one that isn't a compressed file).

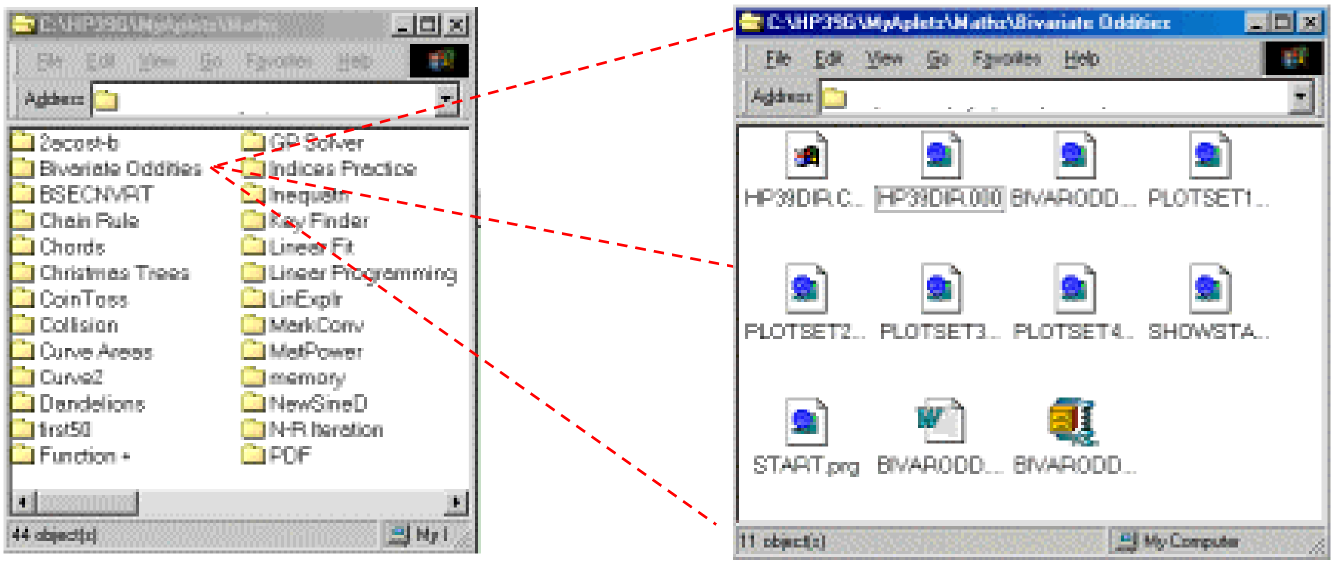
The software that sends the files to the calculator can't work on them if they are inside a compressed file so you *must* expand them before using them.

Calculator Tip

It is **critical** that you decompress each aplet into a separate directory. Each aplet has two special files called HP39DIR.000 & HP39DIR.CUR which always have this same name. Decompressing two aplets into the same directory will cause these two special files from the first one to be overwritten by those of the second. The aplet itself will not be harmed by this but the effect is to render it invisible to the calculator, since these two special files contain information telling the calculator about the aplet.

Organizing your collection

Shown below and right is the contents of one directory in part of my collection.



If you're only going to download a few aplets then organization will not be as important. If you are a teacher or if you are intending to download lots of aplets then you might consider setting up a logical structure of directories to contain them. For example, a teacher might choose to set up a structure containing directories for each of the courses being run, with further directories containing all the aplets which were relevant to that course. ***Again, it must be stressed that each aplet must be in a separate folder!***

Having downloaded an aplet from the web to the computer, we now have the task of transferring it from the computer to the calculator via the HP Connectivity Software. To do this, of course, you need the mini-USB cable that came with your calculator (see page 239) plus some software.

The process of transferring the newly downloaded aplet from the PC to the calculator is exactly the same as it is for an aplet which you have saved to the PC yourself. The instructions for this can be found on page 244.

It is important to realise that most sites contain both aplets and programs. Aplets are stored in the **APLET** view and generally have a **PLOT** view, **SYMB** view and **NUM** view like most normal aplets. Programs do not and are generally less complex and less powerful. A program generally will just ask you for a few values and then display a result and will generally be much less flexible in its operation.

To download an aplet you should be in the **APLET** view when you press the **RECV** button. To download a program you should be in the **PROGRAM CATALOG**. The **RECV** menu will only display things that are appropriate for that view – aplets in the **APLET** view, programs in the **PROGRAM CATALOG**.

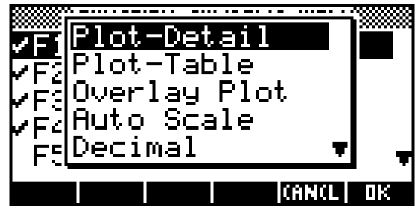
If all goes well, you should find that you presented with a menu which lists the aplet (or program) you are trying to download, together with the word 'Other..'

If all you see is the word 'Other..' then it may mean:

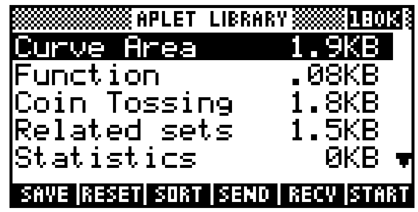
- that you haven't decompressed the aplet properly;
- that the download was corrupted, and the aplet is not readable. This is unlikely but can happen if the HP39DIR files have been corrupted or deleted.
- you are trying to download a program in the **APLET** view or an aplet in the **PROGRAM CATALOG**.

Using downloaded aplets

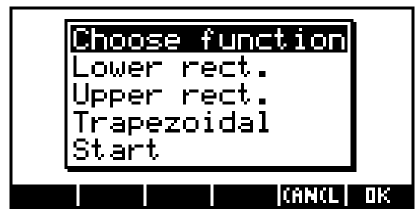
Normally if you press the **VIEWS** key on your hp 39gs or hp 40gs then you will see a list of options which vary according to which aplet is currently active. The **VIEWS** menu for the Function aplet is shown right.



Any aplet that has been created by a programmer, such as the *Curve Areas* aplet shown right, will generally have had its **VIEWS** menu modified by the person who created it.

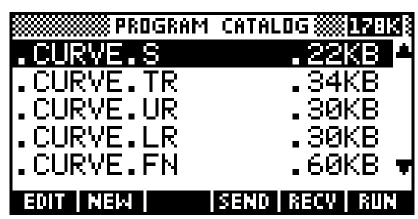


The new **VIEWS** menu is used to control the aplet, offering a series of choices to the user. Usually the intention is that you should work through the choices in the order that they appear. For example, in the menu shown here you would need to *Choose function* before you could use it to investigate areas under the curve using *Lower rect*.

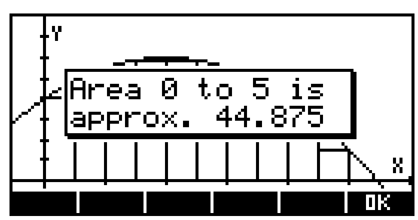


> **Calculator Tip**
> If there is documentation with an aplet then you should read it. It often contains information which is needed for the efficient use of the aplet. It may also have one or more student worksheets which can be copied and used in a classroom.

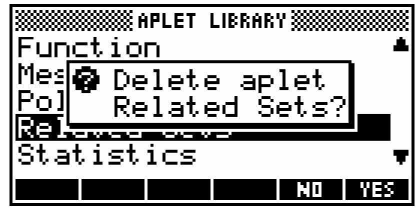
Most of these programmed aplets have 'helper' programs that are automatically loaded into the **Program Catalog**. When you delete the aplet these are not automatically deleted. You must change into the **Program Catalog** and delete them by hand. Left undeleted they will simply consume memory. See the next page.



If you are a teacher then it is highly recommend that you investigate the aplets available. They can enhance your teaching greatly and, more importantly, put the learning process in the hands of the student by letting them control the process via their calculator. It is worth reading the information contained in *Appendix B: Teaching or Learning Calculus* on page 314.

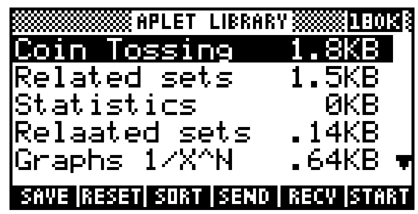


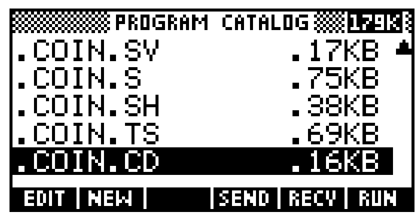
Deleting downloaded aplets from the calculator



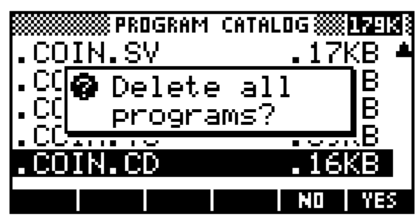
As was mentioned earlier, most of the aplets you download will have 'helper' programs associated with them. These are stored in the **Program Catalog**. If you want more information on these 'helper' programs then read the chapter on programming and, in particular, the **SETVIEWS** command. Apart from curiosity, there is one important respect in which you need to know about these programs and that is when it comes time to delete an aplet. As you would expect, this is initially done in the **APLET** view by highlighting the aplet and then pressing **DEL**.

If the aplet is simple a copy of one of the standard ten, then that is the end of the process. However if the aplet has a set of one or more programs associated with it then those programs will not be deleted and will still be taking up memory. To delete these left over programs you will need to switch to the **Program Catalog** and remove them. Even with 230Kb to play with it is advisable to do this or your memory will gradually be used up.





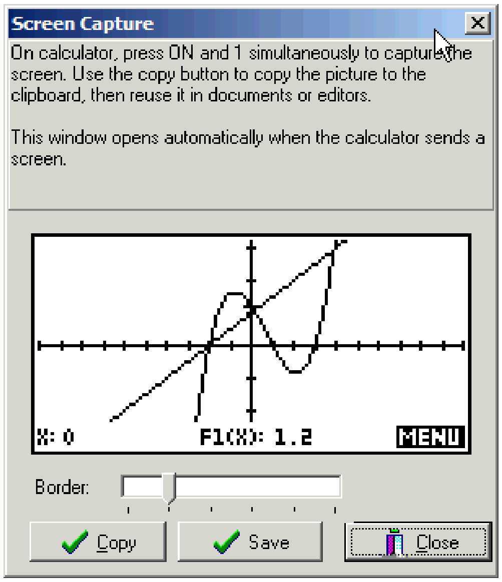
As an example of this, look at the aplet called "Coin Tossing" shown in the **APLET** view right. Looking at the list of programs shown in the **Program Catalog** view, you will see a set of programs which all begin with the letters **.COIN.** The convention encouraged by Hewlett-Packard is to name the programs so that it is fairly obvious which ones belong to which aplet, hence the **.COIN** code for an aplet called "Coin Tossing".



Simply position the highlight on each of the programs in turn and press the **DEL** key. If your calculator only contains one aplet with programs linked, then it is faster to use **SHIFT CLEAR** to delete all programs at once.

The **Program Catalog** view has no **SORT** key. If you store a large number of aplets then you may find that you start to have trouble telling what program belongs and what should be deleted but for the average user this probably will not be a problem. The naming convention should ensure that you are able to work out which is which. If you know the first letter of the name of the program then pressing that letter on the keyboard will take you to it.

Capturing screens using the Connectivity Kit



One of the more useful abilities of the Connectivity Software is its ability to capture images of the calculator screen. These images can be pasted into a document or into a Paint program for further processing. This allows teachers to create worksheets that include images of what the student should see. Students can create reports that include graphs and tables from your calculator.

Any screen can be captured at any moment, even if it is only partially drawn. The capture is triggered by pressing **ON+1**. This means holding down the **ON** key and, while still holding it down, pressing and releasing the **1** key then **ON**. When you do the image appears on the computer.

Once the image is captured it can be copied to the clipboard and pasted into another application. The software also allows it to be saved as a .BMP image file. All the images of calculator screens in this book were captured in this way.

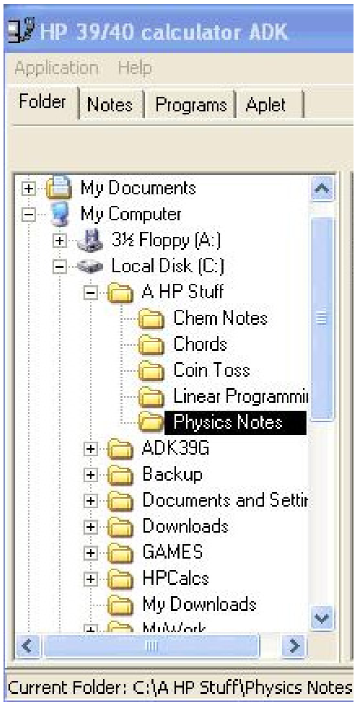
Capturing into the Sketch view

A variation of this capture process is useful if you want to retain the image on your calculator. Pressing **ON+PLOT** stores the image into graphics object G0. This can then be pasted into a sketch using the **VARS** menu. From the **Sketch** view, press **VARS** and then the **HOME** key. Scroll down to the Graphics variables and select G0. Now press **VALUE** and then **ENTER**. The image will be pasted into your sketch and you can then use the drawing tools to edit it. However, using a Paint program is generally easier and quicker.

EDITING NOTES USING THE CONNECTIVITY SOFTWARE

In addition to allowing you to save and retrieve objects from the calculator to the PC, the Connectivity Software will also allow you to edit it once it is on the PC. On previous models this was done using a separate piece of software called the ADK but with the release of new software with the hp 39gs and hp 40gs this ability was integrated into one program.

This software is also used to create and program aplets but this ability is irrelevant to the average user. The average user tends to be a student who is most interested in creating and editing Notes that he/she can take into assessments. Consequently, in this small chapter we will look only at using it to edit Notes and cover programming elsewhere.

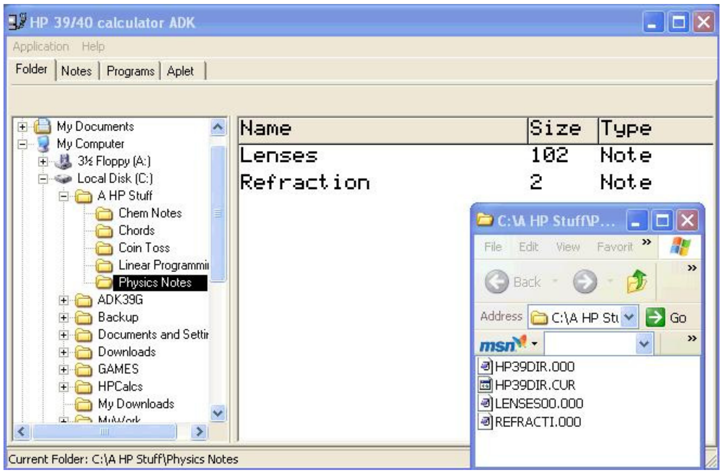


If your Note already exists on the calculator then the obvious first step is to transfer it from the calculator to the PC. This is covered in detail on page 239.

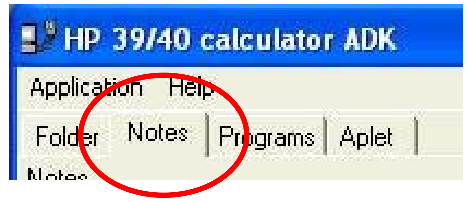
To edit the Note you must select the folder containing it using the directory tree on the Connectivity Software as shown right.

The window pane on the right will then show the contents of the folder. The names are given as they would be displayed on the calculator.

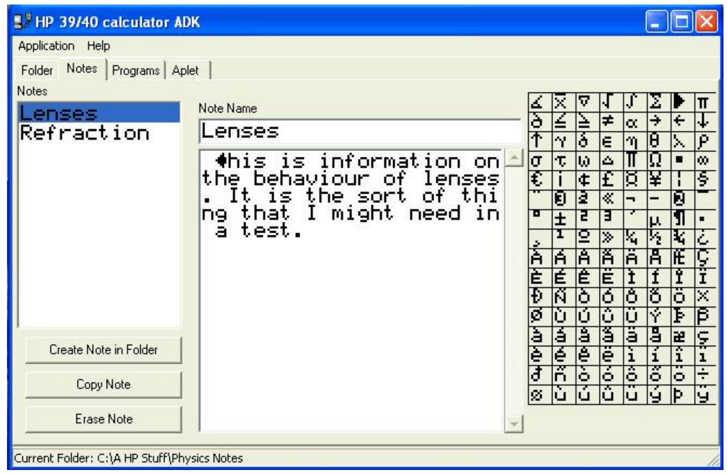
As you can see in the folder, the name on the calculator will probably not the same as the name displayed in the Windows folder.



To edit a Note you need to look at the row of tabs at the top of the editor. Click on the one labeled "Notes" to see any which may be contained in the folder you have already selected.



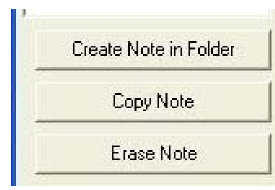
Assuming that you actually have notes in the folder then you will see something similar to the view shown below right.

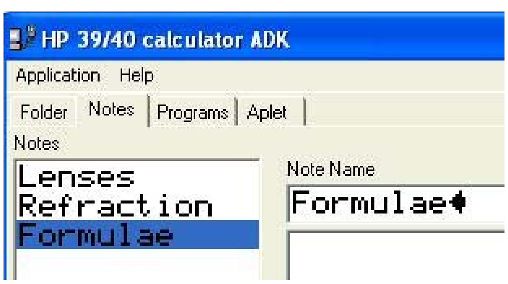


You can now type your text into the Note using the normal computer keyboard. The special characters that are normally available from the **CHARS** view are also available in the editor as you can see in the view shown right.

The changes you make are automatically saved as you type, just as they are on the calculator itself. When you've finished editing you can use the Connectivity Software to transfer the result back to the calculator.

The process so far has shown the editing of a Note which already existed. It is also possible to create Notes on the PC from scratch, to make a copy of an existing Note, or to delete a Note. The key to this is the set of buttons at the bottom left of the screen.

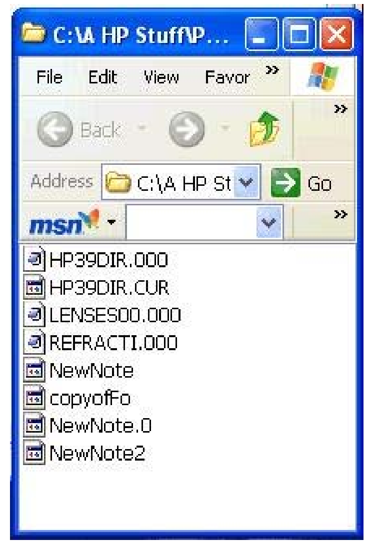




The first button, labeled "Create Note in Folder" will create an empty Note with the title "New Note". You can change that title by typing in the field titled "Note Name". In the example shown to the left the name has been changed to "Formulae".

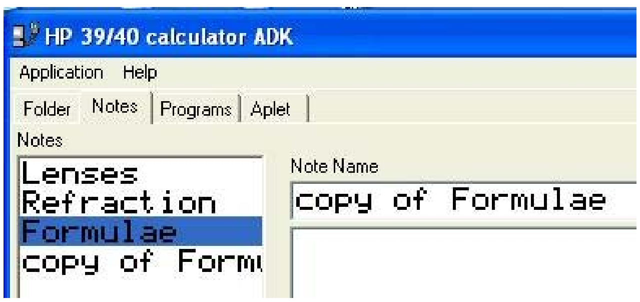
As a matter of interest, this name change for the Note will not change the name of the file in which the Note is recorded on the PC. However, as was discussed earlier, the name on the PC is irrelevant anyway. The name "Formulae" is what will appear on the menu of Notes when you press **RECV** on the calculator.

The names used to record the Notes on the PC are not terribly imaginative, as can be seen to the right.



You must not change these names! They are recorded in the special files HP39DIR.CUR and HP39DIR.000 and the calculator will expect to find them under those names.

Highlighting an existing Note and pressing the Copy button will produce a copy of the existing Note with the title "Copy of ....".

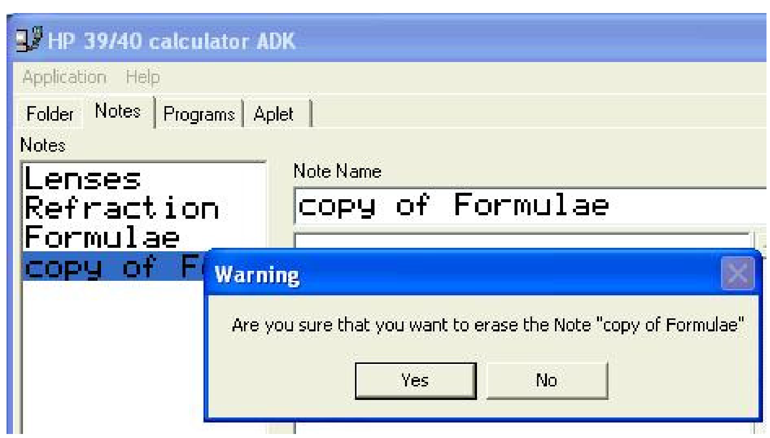


You can then change the title in the normal manner to anything desired.

Calculator Tip

The Connectivity software will allow you to enter a Note title of any length. In practice this will not work. If the name is more than 12 characters long then the calculator will truncate it in the display. For example, "Principles of Conductivity" will appear in the **Note Catalog** as "**Principles o...**".

The final button allows you to delete a Note. Pressing it will result in a pop up dialog box asking you if you are really sure you want to do this.



38

PROGRAMMING THE HP 39GS & HP 40GS

The design process

An overview

Although you can choose to simply create programs, it should be remembered that the whole point of working on the hp 39gs or hp 40gs is to use aplets. Working with an aplet means that you inherit its abilities such as auto setting of axes in the **PLOT** screen and so on. A program shares none of these and must re-create them when needed. This chapter will concentrate on the process of creating aplets which, in addition to their native abilities, possess enhanced powers provided by attached 'helper' programs. Part of this process will involve the creation of these 'helper' programs, and some readers may choose to concentrate solely on that aspect. Whilst this is obviously easier in the short term, the results are far less powerful than the creation of aplets.

The key to the entire process of creating completely new aplets is the **VIEWS** menu and its controlling command function **SETVIEWS**. This function allows you to override the normal behavior of an aplet and superimpose new properties by linking in a set of programs written by you. This is the single most important point in the process and should be kept in mind.

It is mildly deceptive to call these aplets "new", as they always derive from one of the standard ones, but the modification of the **VIEWS** menu means that their final appearance and behavior can be very different to the aplet they derive from.

Essentially the process involves the following stages…

- Choose the parent aplet, based on what abilities you want the child aplet to have;
- Analyze the behavior you require the aplet to have and design the **VIEWS** menu;
- Write the 'helper' programs and attach them to the aplet using the **SETVIEWS** function;
- Add supporting documentation. This last point is often overlooked but in many ways it is as important as the programming itself. Your user must be able to use the aplet or else why did you write it?

Choosing the parent aplet

The first stage in the creation process is to decide which of the standard aplets you wish to make the "parent" of your new child aplet. For some aplets this may not matter, but for others this can be a very important choice. All the abilities of the parent are inherited by the child so the parent choice is crucial if your aplet requires particular abilities. The most commonly used parent aplets are the Function and Statistics aplets, whereas the Quadratic and Trig Explorers would probably not make good parent aplets, since they are specialized teaching aplets without the flexibility of the others.

If your new aplet is going to be concerned with analyzing data then your best choice for a parent would probably be the Statistics aplet. On the other hand if you were planning to write an aplet to teach the behavior of graphs then the Function or Parametric aplets would obviously be best. All the tools of the parent are available to the child, so consider carefully what tools you require.

Working with Software vs Working on the Calculator

When designing aplets you should consider using the software discussed later as it makes the process *far* easier. To use this software you must be able to send to and receive from a computer, and for models before the hp 39gs & hp 40gs this means buying a cable. For the hp 39gs & hp 40gs the mini-USB cable required is included in the package with the calculator.

In this chapter we will begin by creating our first two aplets entirely on the calculator. We will then look at two more examples using the software.

Naming conventions

The process starts by making a copy of the parent aplet and giving it whatever name you want to use for your new aplet. This copy will form the core of your new aplet. Decide also what prefix to use for the programs you will associate with your new aplet. The prefix needs to be recognizably linked to the name of the aplet, so that the user can know which programs to delete when they want to clear the programs out after deleting the aplet from the **APLET** view after use.

For example, an aplet called "Linear Equations" might have a list of programs:

.LINEQ.SV	**.LINEQ.START**
.LINEQ.ENTER	**.LINEQ.DISPLAY**

The next stage is to plan your **VIEWS** menu. The **VIEWS** menu is the controller of your aplet. It pops up when the user presses the **VIEWS** key or at a programming command, and offers a choice of options to the user.

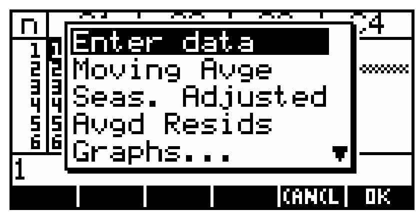
Most of the options in your **VIEWS** menu will be triggers for 'helper' programs you will write, and when the user chooses an option and presses **ENTER**, the appropriate 'helper' program will be run by the calculator. When the 'helper' program terminates the calculator drops into whatever view you as the designer choose.

For example, a 'helper' program might set up axes based on the data entered and then drop the user into the **PLOT** view. Or it might analyze a function and then drop the user into the **NUM** view after setting it up from that analysis.

Planning the VIEWS *menu*

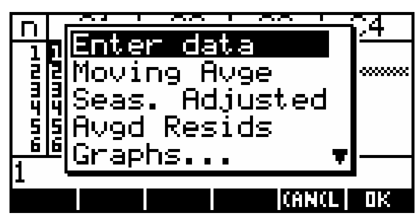
It is crucial to the usefulness of your aplet that you carefully plan the **VIEWS** menu to be clear, concise and user-friendly. Think about the task you are designing the aplet to perform. Break the task down into stages and create the menu to reflect this. Ensure that any task the user would want to perform is included on the menu in the logical order.

It is possible to have sub-menus in the **VIEWS** menu by having your option call a program which then pops up another menu of options. This is usually shown by an placing an ellipsis (…) after the **VIEWS** option, such as the one below right of "Graphs…".

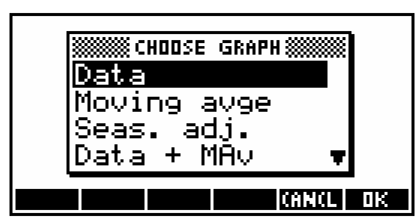


An example of the **VIEWS** menu from an aplet is shown right. The aplet is called "Time Series" and is designed to analyze time series data.

When designing this menu the author has clearly considered the process that a person would normally go through when analyzing Time Series data.

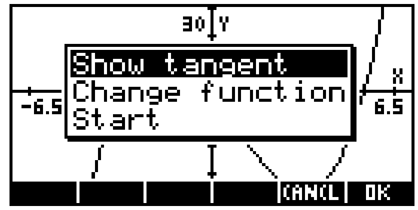


The parent aplet for "Time Series" was the Statistics aplet. This parent was chosen because of the need for the statistical tools it contains. For this particular aplet most of the choices on the **VIEWS** menu trigger a 'helper' program to analyze the data in some way and then drop the user back into the **NUM** view showing the result. Some of the choices drop back into the **PLOT** view to see the data displayed.



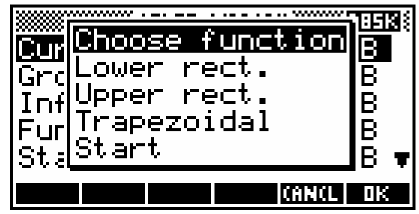
The last option of 'Graphs…' runs a program which pops up another menu, shown right. The reason for doing this is simply to avoid overcrowding on the main menu. Placing '…' (an ellipsis) after the option is a good way to signify that it leads to a further menu.

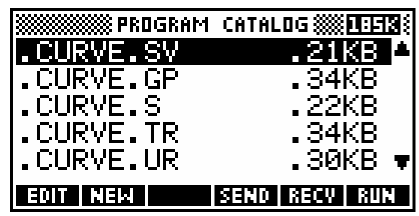
Another example of an existing aplet is shown right. It is called "Tangent Lines" and it draws a tangent line onto a graph and then lets the user move it around, displaying the gradient as it does so.



This aplet has the Function aplet as its parent because of the need to graph functions. It has also had its **VIEWS** menu adapted (see next page) to show the options listed right. In this case the menu has far fewer options because the operation of the aplet is very simple. The reason for the "Start" option is discussed on page 261.

On the right is the **VIEWS** menu for an aplet called "Curve Areas" which finds the areas under curves using various approximations and is used to introduce the definite integral. Because it deals with functions and graphs its parent is the Function aplet. Some of the 'helper' programs for this aplet are shown below. As you can see, their names follow the naming convention discussed earlier.





The SETVIEWS command

The **VIEWS** menu is created by the **SETVIEWS** command. It follows a repetitive pattern of listing a menu option, followed by the name of the program that should run if the user chooses that option, followed by a code number which tells the calculator which view to drop the user into once the program finishes. Part of the job of the 'helper' program is usually to set up this view so that it shows what the programmer wants it to.

You should therefore also think about what you want the user to be looking at once the program they have triggered stops running. Do you want them to be looking at the **PLOT** view - perhaps the option they chose was to draw a graph, with the program being there to set appropriate axes; or the **NUM** view - perhaps we are analyzing data - or should they be looking at the **VIEWS** menu again so they can make another choice?

The syntax for **SETVIEWS** is as follows…

SETVIEWS "Menu line1"; "Program name"; View_No;
"Menu line2"; "Program name"; View_No; (repeated as many times as needed…)
"Menu line3"; "Program name"; View_No: (Note the colon on the final entry)

where View_No is:

0. Home view	11. List Catalog
1. Plot view	12. Matrix Catalog
2. Symbolic view	13. Notepad Catalog
3. Numeric view	14. Program Catalog
4. Plot Setup	15. Views menu item 1 (Plot-Detail if the parent is the Function aplet)
5. Symbolic Setup	16. Views menu item 2 (Plot-Table if the parent is the Function aplet)
6. Numeric Setup	17. Views menu item 3 (Overlay Plot if parent is the Function aplet)
7. Views menu	18. Views menu item 4 (Auto Scale if parent is the Function aplet)
8. Aplet Note view	19. Views menu item 5 (Decimal if the parent is the Function aplet)
9. Aplet Sketch view	20. Views menu item 6 (Integer if the parent is the Function aplet)
10. Aplet Catalog	etc.

The syntax for **SETVIEWS** allows any number of triples. Views 15 onwards vary according to which parent aplet is chosen. The list above assumes the parent is the Function aplet and will be different for other parents.

The convention for the **SETVIEWS** command is to place it in a program with a name of **.NAME.SV** where **NAME** is whatever name you chose at design stage. When you run this program it severs the aplet's link to the normal **VIEWS** menu inherited from its parent and replaces it with the new options. If an aplet is created using the special software then it will probably not have this **.NAME.SV** program. The software creates the **VIEWS** menu in a different way that doesn't require it. This does not affect the aplet in any other way.

The linking performed by the **SETVIEWS** command (or by the ADK) is also important in that it tells the calculator which programs are to be transmitted with the aplet when it is copied via cable or via infra-red link.

Only programs named in the* SETVIEWS *command (or linked by the software) will be transmitted with the aplet by cable or by infra-red.

Special entries in the SETVIEWS command

In addition to the lines which form the menu for your aplet, there are some special entries which are treated differently.

- If you include entries called "Start" or "Reset", then the 'helper' programs associated with those entries will be run when the user presses START or RESET in the **APLET** view. These entries are case sensitive and must appear exactly as shown. See the next page for more on the "Start" option.

- If you include a menu entry which consists of a single space character in double quotes, then the entry will not appear in the **VIEWS** menu, but the program named in the line *will* be transmitted with the aplet. This can be handy if you have a program which is a subroutine. In other words, one which is not directly called from the menu but which is called by other programs which are in the menu. If there is a particular piece of code which is used repeatedly then this allows you to place it within its own program rather than repeating it in each place it is needed.

  An example of this is the **.NAME.SV** program itself. If you attach it to the aplet then it needs to be included in the list in this fashion, since we don't want it to appear on the **VIEWS** menu. Strictly it is not necessary to include this program since, once it has done its job, it would normally never need to be run again but it is usually kept and transmitted with the aplet so that an expert user could modify it if they wanted to.

- An entry which consists of empty double quotes, allows you to access commands which appear on the parent aplet's normal **VIEWS** menu. Since your menu has replaced this normal menu they would usually be lost. For example, the standard menu options of *Auto Scale*, *Plot-Detail* etc. can be included in this way. View numbers 15 onwards are reserved for this purpose.

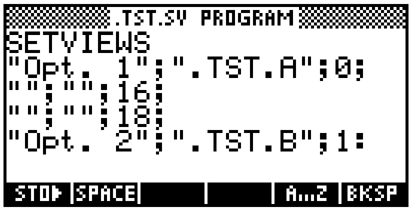
  For example, if your parent aplet was the Statistics aplet in 2VAR mode and you wanted to include its *Auto Scale* command then you would use a view number of 18 since *Auto Scale* is entry number 4 on the normal **VIEWS** menu for the Statistics aplet in 2VAR mode. You need to be quite careful when using this option since the commands like *Auto Scale* appear in different positions for different parents.

  The use of empty double quotes means that the normal name from the parent aplet is used. You can over-ride this and provide your own name.

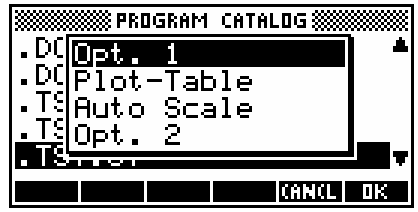
  For example, if the parent aplet was **Function** then the command **SETVIEWS "";".NAME.FRD";18;** will display **Auto Scale**, run the program **.NAME.FRD**. and then perform a normal Auto Scale.

  On the other hand the command **SETVIEWS "MyScale";".NAME.FRD";18;** will do exactly the same thing but the name on the menu will be **MyScale** instead.

Shown below is a **SETVIEWS** program which illustrates this for an aplet with Function as its parent…



producing a menu of…



The behaviour will be:

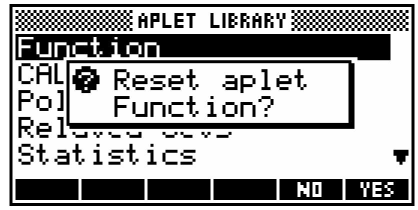
- Choose "**Opt. 1**" Runs program **.TST.A** and then drop the user into the **HOME** view (view 0)
- Choose "**Plot-Table**" Drop the user immediately into the **Plot-Table** view (view 16 for Function)
- Choose "**Auto Scale**" Drop the user into the **PLOT** view after performing a normal **Auto Scale** (view 18 for the Function aplet)
- Choose "**Opt. 2**" Runs program **.TST.B** and then drop the user into the **PLOT** view (view 1)

The 'Start' entry

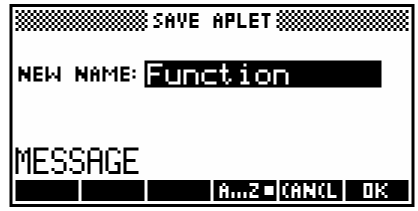
It is a very good idea to include a "Start" entry on your **VIEWS** menu, since it will be automatically run when the user starts the aplet. This allows you to enter pre-set values in variables, to pre-set angle modes, or axes, so that the aplet runs smoothly. You should always bear in mind that the user may make changes to any axes you set or store their own values into variables as part of some other calculation, not realising that you've used them as part of your aplet. The results can be very unfortunate if you don't bear this in mind!

Another important reason for including a "Start" entry is to terminate the "Start" entry with a view number of 7. This means that as soon as the user runs the aplet the **VIEWS** menu (view number 7) will be displayed. This makes the aplet far more user friendly since the controlling menu is the first thing the user will see. Some aplets tend to opt for first displaying the Note view because they include instructions there. I usually opt for the **VIEWS** menu and include instructions in a separate file in Word or PDF format so that they can be printed easily.

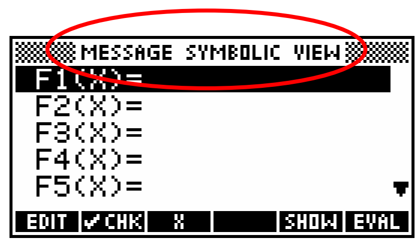
Example aplet #1 – Displaying info



This example uses the **SETVIEWS** command to create a simple (and totally useless) aplet, which illustrates a few of the concepts useful in programming the hp 39gs or hp 40gs. We'll call it the 'Message' aplet and create it as a child of the Function aplet.



Change into the **APLET** view, move the highlight to the Function aplet and **RESET** it. This reset is not necessary but ensures that no settings are left over that may interfere. Now save it under the new name of 'Message' and then **START** this new aplet.



You will find that you are looking at the normal **SYMB** view but for the Message aplet instead of the Function aplet. The title appears at the top of the screen as usual.

Press **SHIFT PROGRAM** to view the **Program Catalog**. Press **NEW** to create a new program and call it **.MSG.SV** (see right, with part of the new program typed in).

Into this empty program, type the following code, obtaining the quotes from the **CHARS** view. When you finish typing, just press **SHIFT PROGRAM** again to exit back to the **Program Catalog**. There is no need to save as this is done continuously as you type.

Spend a moment to go through the code and ensure that you are clear in your own mind the menu it will create, the programs it will run, and the views it will enter after the running of each program. You will be told at a later stage in this example when to run this program and create the menu.

We'll now create the associated 'helper' programs (shown below). Their names/titles are supplied above the code for each one. A short explanation is given. For more information on the various commands see the chapter "Programming Commands" on page 281.

.MSG.2

```
.MSG.2 PROGRAM
ERASE:
DISP 4;"You entered
"N:
DISP 5;" when
prompted.":
FREEZE:
```

ERASE clears the screen, ready to **DISP** a message on lines 4 and 5 of the screen. The calculator then **FREEZE**s waiting for a key to be pressed.

.MSG.1

```
.MSG.1 PROGRAM
MSGBOX "Hello world!
3+4 = "3+4:
```

The **MSGBOX** command is used to display the traditional first message for programmers learning a new language!

.MSG.IN

```
.MSG.IN PROGRAM
INPUT N;"MY
TITLE";"Please enter
N..";"Do as you're
told.";20:
MSGBOX "You entered "
N" when prompted.":
```

The **INPUT** command asks the user for info, displaying a title, prompt and tip and having a default value of 20. You could use the **PROMPT** command instead but **INPUT** is more flexible.

.MSG.FN

```
.MSG.FN PROGRAM
ERASE:
'((X+2)^3+4)/(X-2)'▶F1(X):
→GROB G1;F1(QUOTE(X));0:
→DISPLAY G1:
FREEZE:
ERASE:
→GROB G1;F1(QUOTE(X));1:
→DISPLAY G1:
FREEZE:
ERASE:
→GROB G1;F1(QUOTE(X));2:
→DISPLAY G1:
FREEZE:
ERASE:
→GROB G1;F1(QUOTE(X));3:
→DISPLAY G1:
FREEZE:
ERASE:
LINE Xmin;Ymin;Xmax;Ymax:
BOX 3;3;-2;-2:
FREEZE:
```

This is the most complex of the programs. See right for an explanation.

.MSG.S

```
.MSG.S PROGRAM
MSGBOX "Aplet
starting now":
```

Puts an initial message up.

The command **→GROB** in the program left, stands for "Graphic Object" and creates a GROB from the **F1(X)** expression stored in the **SYMB** view, storing it in the graphic memory **G1**, using the font specified (0, 1, 2 or 3). The reason for doing it this way is to use proper mathematical layout like **SHOW** does. The **→DISPLAY** command then shows it on screen. This is repeated a number of times with different fonts. Finally, a line and box are drawn on the screen. There are more explanations later of all this.

.MSG.SV

```
.MSG.SV PROGRAM
SETVIEWS
"Message 1";
".MSG.1";7;
"Input value";
".MSG.IN";7;
"Message 2";
".MSG.2";0;
"Show func.";
".MSG.FN";7;
"Start";".MSG.S";7;
"Quit";"";0;
" ";".MSG.SV";0:
```

The **SETVIEWS** command is discussed in detail on the previous pages.

Having created all of the programs that make up the aplet 'Message', we can now run the program **.MSG.SV**, severing the aplet's link to its current **VIEWS** menu which was inherited from its parent the Function aplet, and substituting this new menu. *Before you do this*, check that you are still in the correct aplet. Press the **SYMB** key and check that the title at the top still says "**MESSAGE SYMBOLIC VIEW**". If it doesn't show this, then **START** the aplet again to ensure that it is the active one and so the one whose **VIEWS** menu will be changed. This step is critical – you do not want to change the **VIEWS** menu for the wrong aplet!

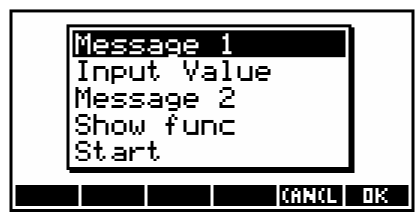
Swap back to the **Program Catalog**, position the highlight on the program **.MSG.SV** and **RUN** the program. Apart from the screen going blank for a moment nothing will appear to happen, but in fact the link to the normal **VIEWS** menu which 'Message' inherited from its parent aplet Function has been severed and a link to the new menu you built in **.MSG.SV** has been substituted. Press **VIEWS** to check. You should find that your new menu appears. Press **CANCL** to exit.

Providing that you have done everything correctly, this is now the end of the process - the aplet is now ready to be run. In the **APLET** view, make sure the highlight is still on the aplet and press **START** or **ENTER** to run it. If you get an error message at any time then you may have to **CANCL** and **EDIT** the program.

When you do this, the aplet will run the program **.MSG.S** which will display a **MSGBOX**.

The line in the **SETVIEWS** command controlling this was:

"Start";".MSG.S";7;



Since the triplet ends with a view number of 7, this means that after the program terminates (when you press **OK**), the **VIEWS** menu will display.

If you choose the option 'Message 1', then this will cause the program **.MSG.1** to be run, displaying the screen on the right. This line in the **SETVIEWS** command also terminated with a view number of 7 so when you press **OK** the **VIEWS** menu will display again.

The program line for this was:

MSGBOX "Hello world! 3+4 = " 3+4:

Items in quotes are displayed as they appear, while expressions outside them are evaluated before being displayed. This means that the 3+4 inside the quotes appears as exactly that, while the one outside is evaluated to 7. Expressions can include variables and calls to functions.

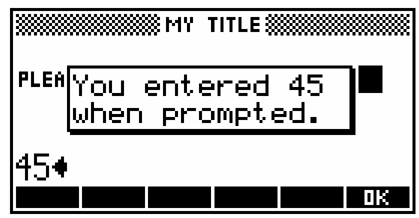
The next option in the menu is 'Input value'. Choosing this option will create an input screen. The statement controlling this was:

INPUT N; "MY TITLE"; "Please enter N.."; "Do as you're told."; 20:

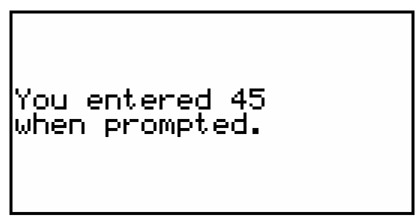
Examine the snapshot on the right and notice the connection between the various parts of the **INPUT** statement and their effect. Note the suggested value of 20, and note also that the prompt of "Please enter N.." was too long to be displayed. See the **PROMPT** command for an alternative that is simpler but less flexible.

When you enter a number into the input screen and press **ENTER**, the next line in **.MSG.IN** will display this value in a **MSGBOX**. When you then press **OK**, the view number of 7 specified in the relevant line of **.MSG.SV** will cause the **VIEWS** menu to be displayed again.

Notice that the input window is still displaying in the background. To stop this happening, you could have included in **.MSG.IN** a line of **ERASE:** , which is a command to erase the display screen. Try editing the program, inserting this line before the **MSGBOX** line, and running it again.



The option of 'Message 2' displays the same message as we saw before, but presented in a different way. The **DISP** command divides the display screen up into 7 lines (1 - 7) on which you can display data.



For example, suppose memory A contained 3.56, then the command:

DISP 3; "The value of A is: " A:

would display the message **The value of A is: 3.56** on line 3 of the display screen.

Notice also that this time when you press **ENTER**, you end up in the **HOME** view rather than in the **VIEWS** menu again. This is not an error. If you look at the line in **.MSG.SV** controlling this option of the menu you will see that its post execution view number was 0 (**HOME**) rather than 7 (**VIEWS** menu) like most of the others. To see the **VIEWS** menu again, press **VIEWS**.

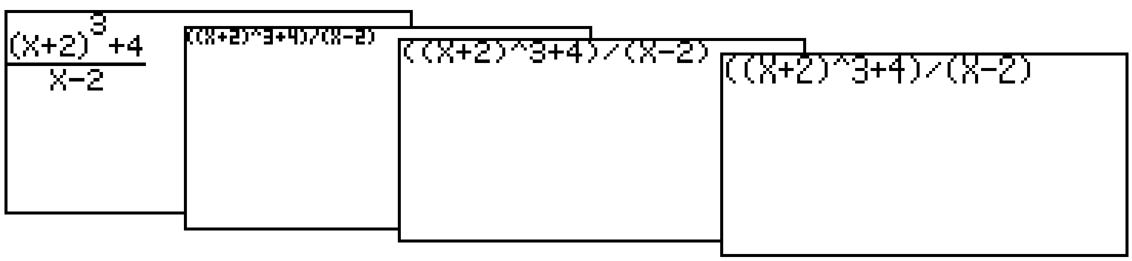
The final option is 'Show function'. The program this runs is a little more complex than the ones shown so far and illustrates a useful technique.

The line: **'((X+2)^3+4)/(X-2)' ▶F1(X):**

stores the expression $\frac{(x+2)^3+4}{(x-2)}$ into the function **F1(X)**.

Notice the way the function is in single quotes so that the algebraic expression itself is used rather than its value when evaluated using the current contents of memory X. If you wanted to graph this function by setting the post execution view number to 1 (the **PLOT** view), then you would need to include the command **CHECK 1:** in order to ✓CHK it or it would not graph. You may wish to edit the **.MSG.SV** and**.MSG.FN** program to try this.

The next lines display the expression using the four options available.



The line: **→GROB G1;F1(QUOTE(X));0:**

converts the expression **F1(X)** into a Graphic Object (GROB). The number at the end, which changes with each repetition, controls the font used to display the function.

The line: **→DISPLAY G1:**

displays this GROB on the screen, and then **FREEZE** freezes the display until a key is pressed.

Finally the **LINE** and **BOX** commands commands are used to draw an oblique line across the screen and a box near the center.

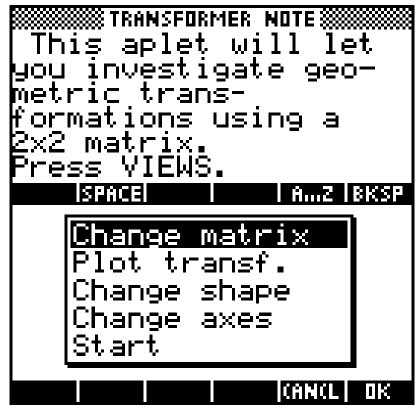
```
LINE Xmin; Ymin; Xmax; Ymax:
BOX 3; 3; -2; -2:
```

Notice the use of **Xmin**, **Xmax**, **Ymin** and **Ymax** in the **LINE** command. These are the minimum and maximum limits of the current **PLOT** view and using them instead of fixed values means that the line would appear the same even if the screen were to be re-sized in the **PLOT SETUP** view. The box, on the other hand, would change size if the screen size were to be changed in **PLOT SETUP**. The **FREEZE** command is needed to ensure that your screen is visible to the user and not immediately replaced by the next view. **FREEZE** resumes execution of the program as soon as any button is pressed.

Note that **SETVIEWS** has a "Start" option and also a final option consisting of a single space in quotes which is simply to link in the **.MSG.SV** program so that it is transmitted with the aplet. It does not appear on the menu but ensures that the **.MSG.SV** program is linked.

Example aplet #2 – The Transformer Aplet

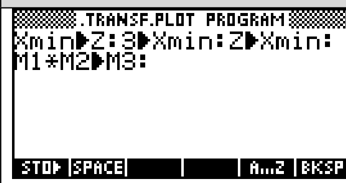
If you haven't already, read pages 234 which explain how to create a copy of the Parametric aplet to explore geometric transformations using matrices. We will now look at using programming to enhance this aplet by automating the process.



Start by highlighting the Parametric aplet and pressing **RESET**. Now **SAVE** the aplet under the new name 'Transformer'. Press **SHIFT NOTE** (not **NOTEPAD**) and enter some explanatory text into the aplet's Note view. You can use the text shown right.

The next step is to create the 'helper' programs for the aplet, including the one containing the **SETVIEWS** command used to create a new **VIEWS** menu for the aplet. These programs are shown on the next page. When you have typed them all in then **RUN** the program **.TRANSF.SV** to create the **VIEWS** menu.

Programs for the aplet 'Transformer' are given below.

.TRANSF.SV	.TRANSF.S	.TRANSF.PLOT
		
This program sets up the **VIEWS** menu to call each of the other programs. It need only be run once at the creation of the aplet, but is attached via the final line so that it will be sent with all the others if the aplet is transmitted. The new user does not have to re-run it: it will never normally be run again unless the menu needs to be modified. The " character can be found in the **CHARS** menu.	This program sets up the required axes using variables from the **PLOT SETUP** view. It then loads the equations and ensures they are **✓CHK**ed and ready for use. Finally it loads the initial values into the matrices. Note that the angle measure is not set since it is not essential. The user may want to use a rotational matrix and can choose their own settings for this.	This program changes the value of Xmin and then changes it back. In the original version the user had to press **PLOT** to force a re-draw. This technique fools the calculator into thinking that the **PLOT** view has changed and therefore forces a re-draw without the need to press a key. It also re-multiplies the matrices in case the user has changed one by hand instead of going through the **VIEWS** menu.

.TRANSF.SHAPE		.TRANSF.MAT

.TRANSF.SHAPE

```
.TRANSF.SHAPE PROGRAM
1▶C:
CHOOSE C;"SHAPE";
"Right triangle";
"Pointer";
"User defined":
IF C==1 THEN
 [[1,2,1,1],
 [1,1,3,1]]▶M2:
ELSE
 IF C==2 THEN
  [[1,2,2,1.5,1,1],
  [1,1,3,4,3,1]]▶M2:
 ELSE
  DO
   EDITMAT M2:
   SIZE(M2)▶L0:
   IF L0(1)≠2 THEN
    MSGBOX"Matrix
must be 2xN":
   END:
  UNTIL L0(1)==2 END:
 END:
END:
SIZE(M2)▶L0:
L0(2)▶Tmax:
M1*M2▶M3:
STO▶ SPACE ▲ PAGE  A…Z BKSP
```

This program uses the **CHOOSE** command to offer a list of options. Note the need to pre-load a value into **C**. This value determines which option is initially highlighted when the menu appears. If a list has only three options but the highlight is set to some other value than those three then it can crash the program.

(continued…)

Since the default contents of any variable is zero and there is no zero'th option on a list this means a program bug waiting to happen unless you preset the value.

Options 1 and 2 load preset matrices while option 3 allows the user to edit their own. Note the check to ensure the matrix they entered has a valid size. The number of columns is then extracted and used to reset the value of **Tmax**. The new image matrix is also recalculated.

Note: The indenting used is not required and is there simply to make the program easier to read. The amount of memory take up by a few extra spaces is minimal but well worth it in terms of readability.

.TRANSF.MAT

```
.TRANSF.MAT PROGRAM
MSGBOX"Enter the 2x2
matrix.
Press OK when
finished.":
DO
 EDITMAT M1:
 IF SIZE(M1)≠{2,2}
  THEN
  MSGBOX"Matrix must
be 2x2 only.":
 END:
UNTIL SIZE(M1)=={2,2}
END:
M1*M2▶M3:
STO▶ SPACE ▲ PAGE  A…Z BKSP
```

This program puts up a message instructing the user and then allows them to edit the transformation matrix in **M1**. The size of the matrix is checked to ensure it is 2x2, with the **DO…UNTIL** loop ensuring that the user cannot exit without a valid matrix entered.

Assuming that you have **RUN** the **.TRANSF.SV** program to create the new altered **VIEWS** menu then you can now test the aplet. Don't forget that you must **START** the aplet first to ensure that the new **VIEWS** menu is attached to the right aplet. Its operation should be familiar to you if you have already examined its 'cousin' on page 234.

Designing aplets on a PC

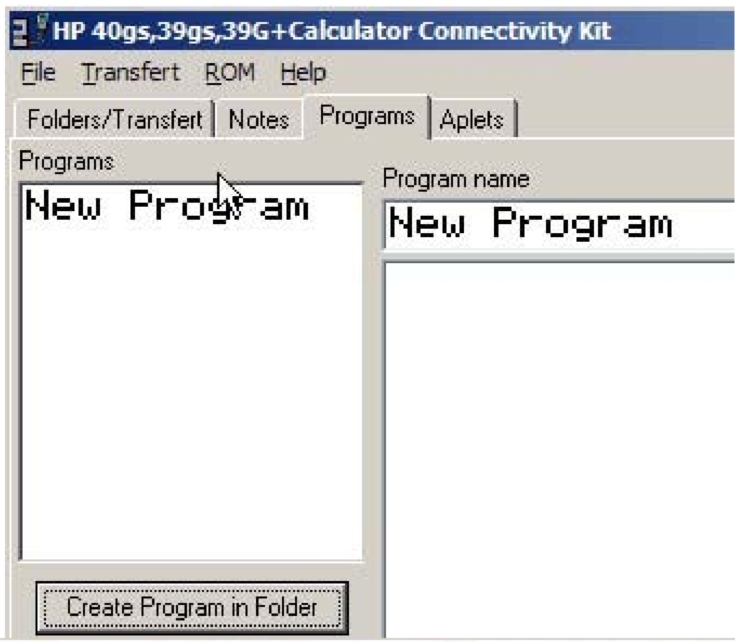
> ***Please note***
> The software used on the PC to edit and create Notes, programs and aplets was in the process of being written at the same time as this book. Consequently the explanations given here may tend to be a little vauge. Screen images and explanations may be different on the final version you are using. However the process should be substantially correct. At the time this book was being written the sofware was called "The Connectivity Kit" but this too may have changed.
> Look for the latest version on HPs website or on *The HP HOME view* (at http://www.hphomeview.com).

In the next example we will use The Connectivity Kit to create small program and then to re-create the same 'Transformer' aplet used in example 2. This second example will allow us to concentrate fully on how to use the The Connectivity Kit rather than the programming task since the programming has been discussed in great detail earlier.

Example program "Log X (base b)"

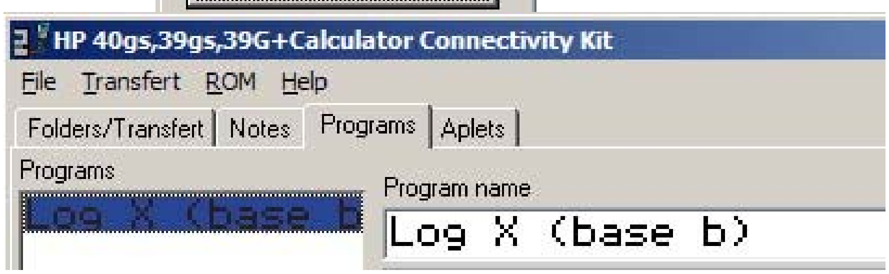
This is a small program that will find the log of any number to any base using the change of base law. Clearly this is not terribly useful but it does illustrate the process in a simple way.

Begin by creating a folder to hold the program or choosing an existing one if that's what you want. Start the Connectivity Kit and select the folder in the initial Folder/Transfer tab.

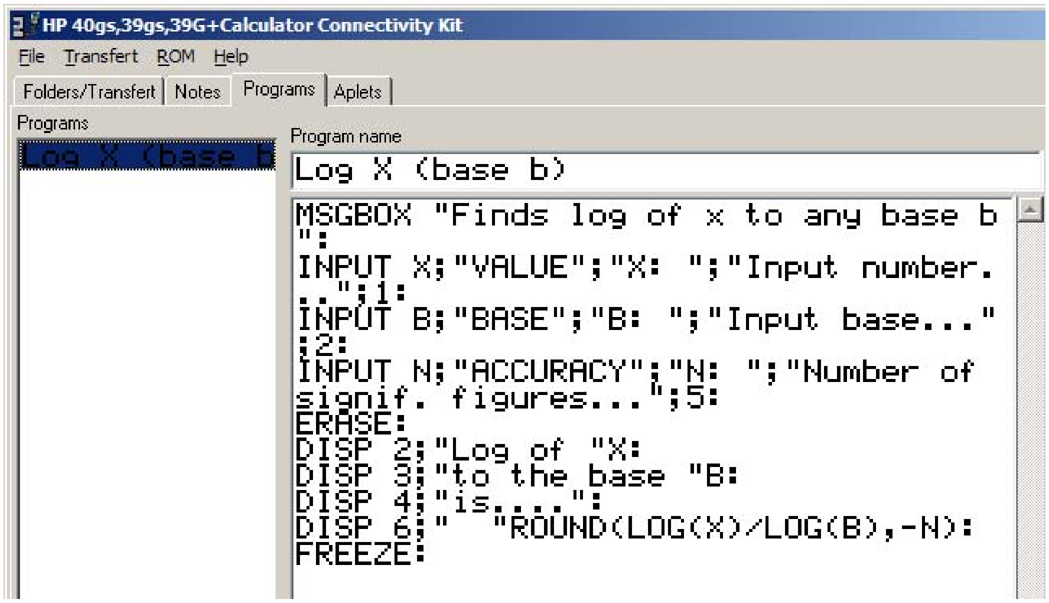


Change into the Program editing window by clicking on the Programs tab below the menu bar and then create a new program by clicking on the "Create Program in Folder" button. The result is shown right.

Enter the name you want to use into the "Program Name" field.



Type the code for the program into the code window.

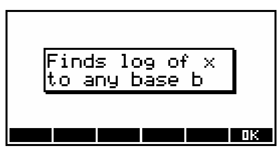


At the time this book was written there was some debate going on over whether the code should be saved automatically as you type or whether it was better to have the user click on a "Save" button.

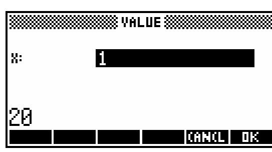
So… if there is a "Save" button or an option on the File menu of "Save" then use this now to save your code. If there's no such option then go on to the next step.

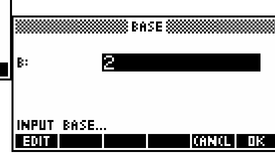
Change back to the Folders/Transfers and download your program to the calculator for use. The result is shown in operation below.

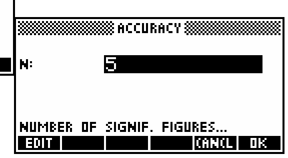
The result of the **MSGBOX** statement is:



This is followed by a succession of **INPUT** statements to obtain the value, the base and the required accuracy. Notice the way that the **INPUT** statement allows you to put a title, a prompt and a more detailed explanation on the screen to help the user see what is required of them.

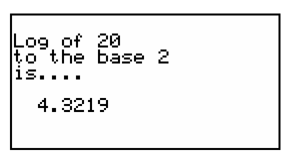






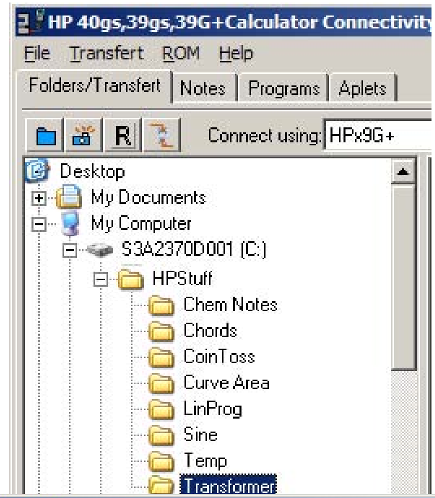
The result is shown using the **DISP** command, which divides the screen up into seven lines of text. An alternative would be to use another **MSGBOX** statement such as

MSGBOX "Log of "X" to the base "B" is: "ROUND(LOG(X)/LOG(B),-N):

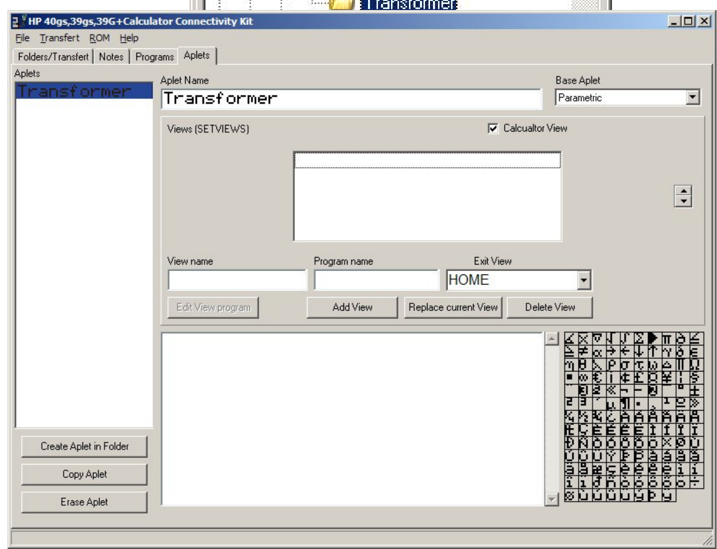


Of course if the user enters a negative number this program will not work correctly, displaying a complex result. A caring programmer would insert an error-trap into the program: lines to check that the user has entered a positive value before performing the calculation, displaying an error message if they have not.

Example aplet #3 – Transformer revisited



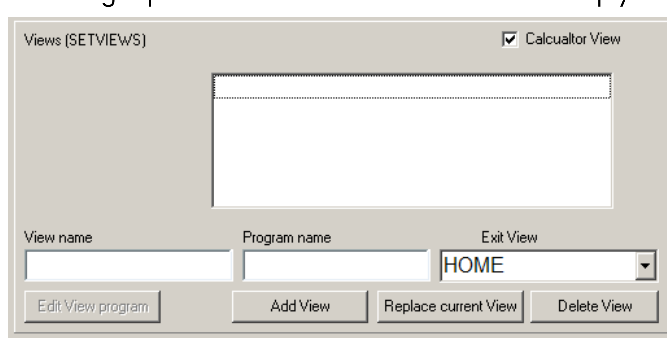
Run the Connectivity Kit and use the File menu to create a new folder called "Transformer", and highlight that folder to hold your aplet. In my experience it is a very good idea to store each aplet in a separate folder but this is not strictly necessary.



Change to the Aplet tab view to see the view shown right. Create a new Aplet, selecting a Base Aplet of "Parametric" and using the appropriate button. Change the name of the aplet to "Transformer" as shown.

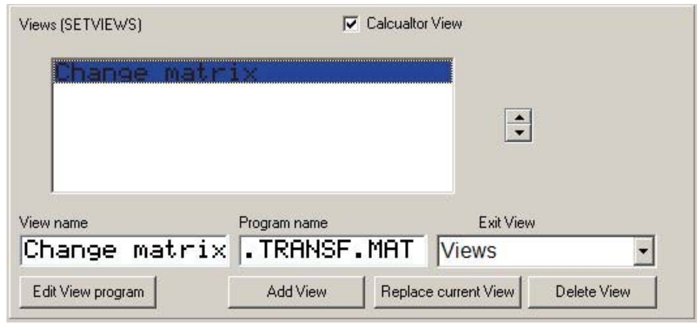
The aim now is to create the **Views** menu, adding code to each view as we go. If you are not sure what the **Views** menu is then you should go back and read the information starting on page 255 before proceeding.

This is done using the portion of the screen shown to the right. As was explained on page 259, the **SETVIEWS** command works using triplets of information and this screen simply automates this process. For each entry you supply a **View Name** (the text that is to appear in the menu), a **Program Name** (this program runs when they choose this option) and an **Exit View** (the view that you are placed in when the program finishes running).



As you enter each triplet of information you should press the **Add View** button to add it to the menu. Other buttons are provided to delete an entry or to edit it by replacing the currently highlighted one with yours.

As you enter each triplet, the boxes will blank ready for the next menu item to be added. You can construct the entire menu at one time OR you can edit the code for the program before proceeding. In many ways it is better to design the entire menu structure before beginning to code but that may not be the way you prefer to work.



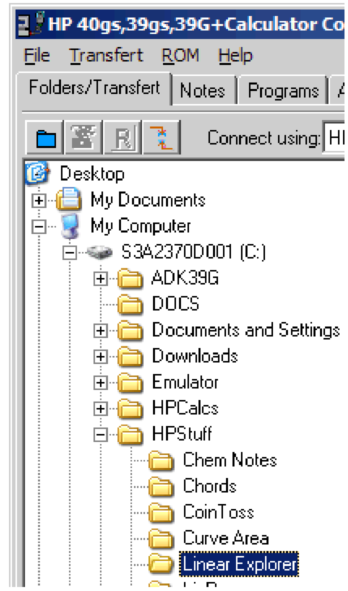
In the window shown above right you can see that after the view triplet has been added to the menu the **Edit View Program** button on the far left is enabled. Clicking on this changes the focus from the Aplets tag to the Program tag, allowing you to enter the code for the program.

You should practice this process now by re-creating the Transformer aplet which was used as Example #2 on page 268. You will find the code listed on the pages following 268.

Please note that if you open the folder in which you have created your aplet you will find the aplet, the programs and also two files called HP39DIR.CUR and HP39DIR.000. Information on these files, which are created automatically by the programming utility, can be found on page 246.

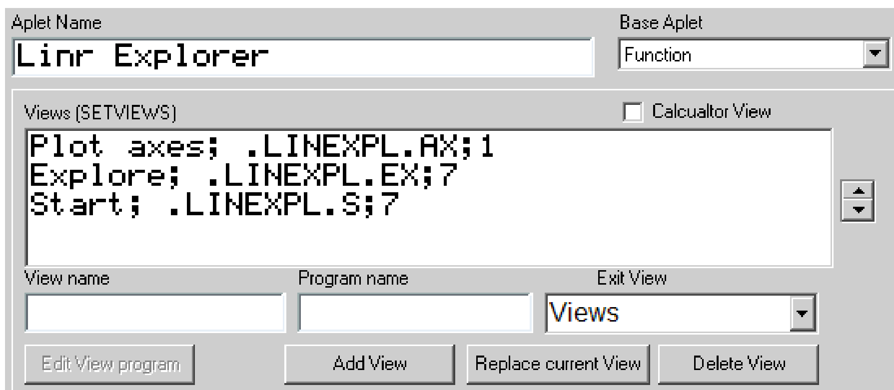
Example aplet #4 – The Linear Explorer aplet

If you would like more practice in using the programming utility then you may wish to use it to create this final example, which is a very useful teaching aplet called "Linr Explorer". The name would be better as "Linear Explorer" but names of more than 14 characters will not display properly in the calculator's **APLET** view. This aplet will be somewhat similar to the Quad and Trig Explorer aplets, except that it will explore linear equations. Its parent is the Function aplet.

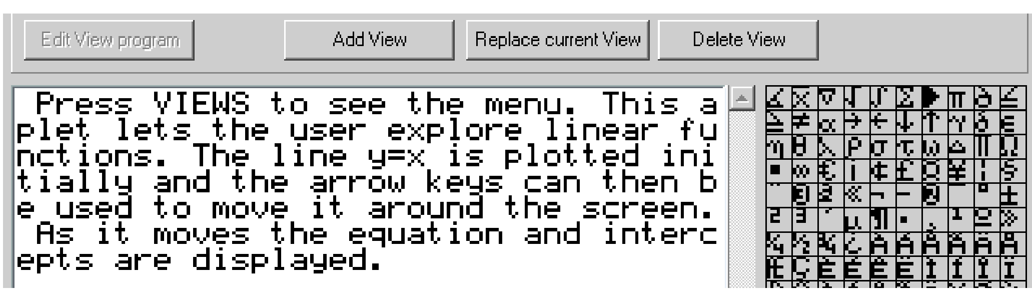


As with the previous example, begin by creating a directory to hold the aplet and then run The Connectivity Kit. Select it in the directory window so that any files crated will be saved there.

Create an aplet with a parent/base aplet of "Function" in the same way as the previous example.



If desired, you can add text to the aplet's Note view (see page 217) in order to give the user instructions. Although this is a good idea, I find in practice that very few users read it! The grid to the right of the Note entry area allows you to use special characters from the **CHARS** view. As can be seen in the example below, there is no attempt to place line breaks at the end of words. The view on the calculator will certainly not match that on the PC so why bother?



If you have done this correctly then your **VIEWS** menu have three entries shown right when it is transferred to the calculator.

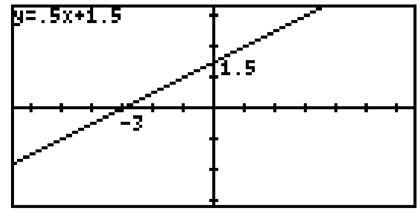
The text for the 'helper programs' associated with each menu entry is shown below:

```
.LINEXPL.AX PROGRAM
MSGBOX"Use PLOT SETUP
and replot if
different axes are
wanted":
1▶G:
'Ymax+1'▶F1(X):
CHECK 1:
```

```
.LINEXPL.S PROGRAM
0▶G:
'Ymax+1'▶F1(X):
CHECK 1:
-6.5▶Xmin:6.5▶Xmax:
-3.1▶Ymin:3.2▶Ymax:
```

```
.LINEXPL.EX PROGRAM
IF G==0 THEN
 MSGBOX "You must'Plot axes' first":
ELSE
 1▶PageNum:
 PLOT→ Page:
 MSGBOX"Use →←↓↑ to adjust line.":
 MSGBOX"Press ENTER to finish":
 1▶M:0▶C:
 DO
  →DISPLAY Page:
  IF C<0 THEN
   DISPXY Xmin;Ymax;1;"y="M"x"C:
  ELSE
   DISPXY Xmin;Ymax;1;"y="M"x+"C:
  END:
  DISPXY .3;C;1;C:
  IF M≠0 THEN
   DISPXY -C/M;-.3;1;ROUND(-C/M,2):
  END:
  LINE Xmin;M*Xmin+C;Xmax;M*Xmax+C:
  GETKEY K:
  CASE
   IF(K==25.1)THEN C+.5▶C:END
   IF(K==35.1)THEN C-.5▶C:END
   IF(K==34.1)THEN M+.5▶M:END
   IF(K==36.1)THEN M-.5▶M:END
  END:
 UNTIL K==105.1 END:
END:
ERASE:
```

It will probably be easier to understand how the aplet works if you see it in action first so you may wish to download the aplet from The HP HOME View and try it. Alternatively, simply read the summary below.



The first option on the **VIEWS** menu plots a set of axes and must be chosen first. Once this is chosen the user can then choose the second option on the **VIEWS** menu to explore the equation of a line. As you press the up/down arrows the line moves vertically by changing the value of 'c' in y = mx + c. Pressing the left/right arrows twists the line by changing the gradient 'm'.

On the pages following we will examine the code in detail, as it illustrates many highly important techniques.

Analysing the aplet

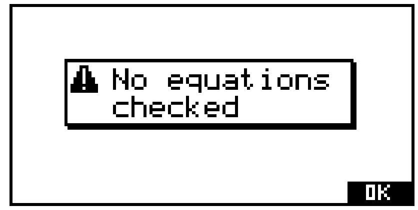
Since it is the first program run we will look first at the program **.LINEXPL.S**.

Because the **VIEWS** menu entry for this is 'Start' it will be run automatically when the user presses **START** in the **APLET** view. This is crucial because we can use this to set the initial values of various variables. Including a **Start** option on your **VIEWS** menu is a very good idea. The single most important feature of this particular entry is the fact that the **Exit View** was set to 7 (the **VIEWS** menu). This means that when the user presses **START** the first thing they will see after this code executes is the **VIEWS** menu.

The variable **G** is being used here as a flag. A 'flag' is a programming term for a variable that is used to tell whether or not some task has been done or some setting has been set correctly. Often the flag is set to 1 to indicate true or 0 to indicate false. Before the 'Explore' option can be used we must be sure that the axes have been plotted, and this will be done by checking the value of **G**. Zero will mean 'unready' and one will mean 'ready'. When the 'Plot axes' option is run the value of **G** will be set to 1 and this will be checked before allowing the user to run the 'Explore' option.

By setting it to zero in the program which is run when the user presses **START**, we ensure that the correct value is going to be in **G** initially. If this check is not done then the code used later in **.LINEXPL.EX** will cause the program to crash, which is obviously not something we want. The axes are also set back to the default settings in case the user has changed them. It is possible to change them once the aplet is running but initially we want them set to particular values that we know will work well.

The second and third lines insert a function into **F1(X)**. This can only be done, of course, if the parent aplet is **Function**. If you do this when the parent is another aplet then the code will still execute but the function will be inserted into the real **Function** aplet!



The reason for inserting this particular function is that we need a function when the axes are plotted or the normal error message will be displayed (see above right). This is undesirable because it will confuse the user. On the other hand we need *blank* axes for later use in exploring, so we use a trick – we set the function being inserted to be 'Ymax+1'. This is a constant linear function, a horizontal line, which is guaranteed to be off-screen for the entire domain no matter what axes are used vertically. Clever, eh?

The next program code we will look at belongs to the 1st option on the **VIEWS** menu of 'Plot axes'.

```
.LINEXPL.AX PROGRAM
MSGBOX"Use PLOT SETUP
and replot if
different axes are
wanted":
1▶G:
'Ymax+1'▶F1(X):
CHECK 1:
STO▶ SPACE ▲ PAGE  A…Z BKSP
```

A message is first given to the user of how to proceed if they want to use different axes. The flag value of **G** is then set to 1 so that the next program can tell that the axes are ready to use. The function is also re-entered in case the user has changed the **SYMB** view. Users have an annoying habit of changing things so try to allow for this in your programs by making them fool-proof.

The next program below runs when the user chooses the second menu option of "Explore", and illustrates a very important technique. A copy of the **PLOT** view is stored in the aplet's sketch view and then retrieved and modified using the various graphics commands. The program is broken into parts for discussion purposes.

```
.LINEXPL.EX PROGRAM
IF G==0 THEN
 MSGBOX "You must'Plot axes' first":
ELSE
 1▶PageNum:
 PLOT→ Page:
 MSGBOX"Use →←↓↑ to adjust line.":
 MSGBOX"Press ENTER to finish":
```

The reason for the "**IF G==0 THEN**" is to check that the blank axes have been plotted and are available for use. If not then the user receives a message to tell them what to do and the remainder of the program is bypassed using the **IF…THEN…ELSE** statement. Trying to capture a **PLOT** view that doesn't exist is a major error and will result in the program crashing abruptly. It is possible to allow for errors like this using the **IFERR** statement but in a teaching example like this it makes the code more difficult to follow.

Still referring to the code on the previous page, you will see that it refers to **PageNum**. The sketches in the calculator's **SKETCH** view are numbered 1, 2, 3…etc. Sketch number 1 is always present but after that only sketches that have been created are available and the program will crash if you try to access one that does not exist.

The aplet variable **PageNum** is the pointer to the sketch you want and the actual sketch page itself is called **Page**. Thus the two lines after **ELSE** are telling the program to store the **PLOT** view into the first page of the **SKETCH** view using the command **PLOT→**. This command stores the **PLOT** view into whatever graphics variable you specify. In this case into **Page**.

The **PLOT** view *must* exist before this can be done or the program will crash. This is the reason for setting up the flag **G** discussed earlier – by doing that we ensure that this section of code only runs if something has been plotted. If you run the program and then later change to the **SKETCH** view you will be able to see this stored image. Finally, the user is presented with two messages which tell them what to do.

The next section contains the code which performs the work in the aplet by setting up a loop which repeats until the user presses the **ENTER** key to terminate.

```
1▶M:0▶C:
DO
 →DISPLAY Page:
 IF C<0 THEN
  DISPXY Xmin;Ymax;1;"y="M"x"C:
 ELSE
  DISPXY Xmin;Ymax;1;"y="M"x+"C:
 END:
 DISPXY .3;C;1;C:
 IF M≠0 THEN
  DISPXY -C/M;-.3;1;ROUND(-C/M,2):
 END:
 LINE Xmin;M*Xmin+C;Xmax;M*Xmax+C:
```

The first line before the loop begins assigns initial values to the variables **M** (the gradient) and **C** (the y-intercept). The **DO…UNTIL** loop which follows (partly in the next section of code) loops through the code within it until the **ENTER** key is pressed.

Within the loop, the previously stored **SKETCH** view is transferred from storage to the display using **→DISPLAY**. The **→DISPLAY** command means "transfer to the display screen". The equation of the current line is then displayed in the top left corner using the **DISPXY** command which allows you to write text onto the screen. Two versions are needed to avoid an expression like "y=2x+ -1" and write instead "y=2x-1".

The **DISPXY** command is a hugely useful command to programmers. It appears in the *Prompt* section of the **MATH** menu. It allows you to place a string of text at any position on the screen using two different fonts and has the syntax:

DISPXY <x-position>;<y-position>;<font#>;<object>:

For example, suppose **M=2** & **C=3**. Then the command `DISPXY Xmin;Ymax;1;"y="M"x+"C:` will display the text "y=2x+3" using font number 1 (small) at the top (Ymax), left (Xmin) of the screen. Positions are given in terms of the current screen coordinates as defined in the **PLOT SETUP** view.

The next line places a label on the y axis (offset slightly) to mark the y-intercept. A check is then done to see if an x-intercept exists and, if it does, a label is placed to mark it. Any labels off the edge of the screen will be ignored by the calculator so you don't need to check for that in the program. This is a very nice feature of the HP. Many calculators have problems with objects plotted off the screen, particularly if they start on the screen and end off the edge. The HP does not and this simplifies your programming task considerably at times.

Finally the line itself is drawn. Even though part of the line extends off the screen there is no problem - the excess is clipped by the calculator.

The next section of code below waits until the user presses a key (**GETKEY**) and stores the key's code into the variable **K**.

```
   GETKEY K:
   CASE
    IF(K==25.1)THEN C+.5▶C:END
    IF(K==35.1)THEN C-.5▶C:END
    IF(K==34.1)THEN M+.5▶M:END
    IF(K==36.1)THEN M-.5▶M:END
   END:
 UNTIL K==105.1 END:
END:
ERASE:
STO▶ |SPACE| ▲ PAGE | A…Z |BKSP
```

A **CASE** statement is then used to check for the use of the arrow keys. Notice the lack of colons (:) after each **END** in the **CASE** statement. Normally every command ends in a colon but the code within a **CASE** statement is one of the few times where this is not true.

If the left or right arrows have been pressed (keys 34.1 or 36.1) then the line is 'twisted' by changing the value of **M**. If the up or down arrows have been pressed (keys 25.1 or 35.1) then the line is raised or lowered by changing the value of **C**. See the manual for more information on the **GETKEY** and **CASE** commands and on the meanings of the key values.

The final check in the line **UNTIL K==105.1 END:** is to see if the user has pressed the **ENTER** key. If so then the loop will terminate and the screen will erase. If not then the loop begins again with the new line being displayed. On termination of the program the **VIEWS** menu will display again, because we chose this when designing the aplet in the ADK.

This aplet illustrates most of the commonly used programming techniques. If you would like to gain experience then I suggest that you do as I did when I first began programming the HP - download aplets and pull them apart to see how they work.

If you would like to further enhance this aplet then try the following:

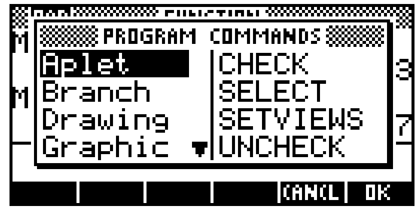
- change the order of the code so that the labels are drawn after the line, thus ensuring that the text is never obscured by the line.

- add a new variable **D** to allow the size of the increment to change. Set an initial value of **D** of 0.5 at the same point as the values of **M** and **C**. Then change the lines in the **CASE** stataement so that **D** is added/subtracted instead of 0.5. Add two more **IF**s to the **CASE** statement so that if they press '+' the increment doubles, and if '-' then it halves. You will also need to add another message box line telling them about this. Finally, add a line to display the current increment size at the top right of the screen using the **DISPXY** command. Be careful that your display does not go off the screen.

The explanation so far should help you in understanding the programming process on the hp 39gs & hp 40gs. The aplet structure is well designed and, if you take advantage of the **VIEWS** menu structure, offers easy creation of complex and very powerful teaching aplets. Certainly what has been discussed here is enough that a programmer will be able to make a start without some of the errors that I made.

39

ALTERNATIVES TO HP BASIC PROGRAMMING



The hp 39gs and hp 40gs are supplied with a simple and easy to use programming language called HP Basic. This language is compiled rather than interpreted, which means that when you run a program it is translated into machine code before it is run. This saves time when running but causes a slight pause the first time any program is run while the translation process goes on. After the first time the compiled version is remembered and re-used unless you edit the program. Other calculators often use an interpreted language where each line is compiled only as it comes up for execution. There are advantages and disadvantages to both methods.

The drawback of the HP Basic language is that it is relatively slow to execute when compared to a modern computer and also that it is missing certain features commonly found in modern languages. The most significant of these is string handling. Unfortunately there is no way to input a string (a set of characters), nor can you work with strings or store them. In other words you can't, for example, ask the user what their name is and then display messages addressed to them by name. This can be very frustrating to those used to other languages but there are generally ways to work around this. The most common is to pop up a menu with options from which the user can choose. The problem with this is that you have to know in advance what the options are going to be – you can't easily re-program them on the fly based on user input.

The sRPL programming language

The speed issues and the lack of string handling, mean that programs such as games are not easy to create. Some programmers choose to work instead in an alternative language called sRPL. For those who have some programming background, this is an assembly code language. If you're not sure what this means, then basically it means working at a language level where you are writing commands that are executed directly on the chip rather than being compiled & translated by the calculator first. HP Basic is a language that is not far removed from the way that humans think (mathematically at least). sRPL is not like this in that it is written in the way that the chip 'thinks' instead. But it does offer both tremendous speed and string handling.

Unfortunately this means that the commands in sRPL are far harder to use and far more fiddly and verbose. A program that takes five lines in HP Basic might take 50 lines or even more in sRPL. However it is quite likely that those 50 lines will execute fifty to sixty times as fast as the five lines of HP Basic. The drawback is that they will also have a much greater risk of containing a bug – the compiling process of HP Basic will spot most simple bugs but sRPL has no such filter. It just does what you tell it to even if this means crashing the entire calculator. It is an very powerful language to use but you need to be very careful or your code will regularly crash the calculator. Additionally, while a crash in HP Basic will simply cause the program to stop unexpectedly, a crash in sRPL may well cause the calculator to lock up or reset with complete loss of user memory.

For those interested in learning HP Basic a series of tutorials can be found on the author's website at http://www.hphomeview.com. Look for the Help page and find the programming section.

The HPG-CC Programming language

The hp 39g+ was the first of this family of calculators which didn't use the *Saturn 5* as its ROM chip. Up to that point the HP38G, HP39G & HP40G had all shared the same chip along with others in the HP48 family. However, supplies of the chip ran out world wide in about 2003 and so the hp 39g+ took a different route. Instead it uses an extremely fast ARM processor (slowed down slightly to save battery power) and simply emulates the Saturn 5. That is, it runs a special program on the ARM processor in machine code that 'pretends' to be a Saturn 5 chip. This virtual chip then runs the calculator's operating system. The calculator is completely unaware that it is not using a Saturn 5. This may seem a strange thing to do but it avoided having to completely re-write the operating system, which would have been very expensive. To the user there is no difference except for a considerable increase in speed. The 39g+ was on the order of 10 times as fast as the 39G. To the programmer, however, there is potentially a BIG difference.

Because the underlying chip is a fast ARM processor it is possible to write code which completely bypasses the virtual chip and runs directly on the underlying ARM chip. This results in incredibly fast code, sometimes on the order of 100 times faster. It also means that you can use other high level languages to write your programs, rather than being confined to HP Basic. In particular, you can use a version of C called HPG-CC created by a very talented group of programmers.

The drawback of this is that by bypassing the virtual chip, you lose access to ALL the abilities of the calculator itself - it's graphing facilities, it's built in functions and so on. Essentially you are operating on a bare chip and having to create everything yourself. It is possible to access some of these via back door methods but they are not simple. In addition, the HPG-CC language itself provides some of these abilities, such as writing to the screen and these abilities are being added to all the time.

Warning!!

A drawback of HPG-CC is that it is possible to write code that will completely crash your calculator! By this I mean not just to the point where you have to do a hard reboot and lose the user memory but to the point where you overwrite the calculator's operating system and can't even reboot. A programmer might also write a loop in their program that took over the ARM chip and monopolized it to the extent that you could not interrupt it using the normal keyboard reset.

This language is *not* for the beginner programmer. Unless you have some experience with using C in another, safer, environment I would *very* strongly suggest that you not consider using it! Windows for example, has safeguards built into it that prevent a program from damaging the operating system itself. The calculator does not.

The good news is that even if you do manage to overwrite the operating system you can always download a new copy from Hewlett Packard's website and re-install it into the flash ROM (see page 284). This is not hard to do although it would obviously be annoying. A loop that can't be reset using the normal keyboard methods can still be interrupted using the physical reset button located inside the hole in the back of the calculator (see page 42), since this button actually interrupts the power from the batteries and so will 'kill' any program. If necessary, remove the batteries.

The HPG-CC language was originally developed for use on the hp 49g+, which is a more sophisticated graphical calculator aimed at university and professional use. Support for the hp 39gs & hp 40gs was added later. At the time of writing this information in early 2006, HPG-CC had reached Version 2.0 but subsequent to this version the support for the hp 39g+, hp 39gs & hp 40gs had been discontinued. This is not to say that it won't work on those calculators, just that the developers are no longer fixing bugs discovered by programmers on those models or taking those models into account when developing new features. This is basically due to lack of interest from programmers - the developers can't really be expected to continue to support a model that no-one's programming on. I myself have written a few small programs in it but just didn't have the time to learn it fully. If the language becomes popular on the new models then this may change.

To download this language it is best to go to the Help page on the author's site The HP Home view since the link on that page will be to a relatively recent version. The latest versions contain a Windows interface that allows you to program more easily (the original versions were DOS based).

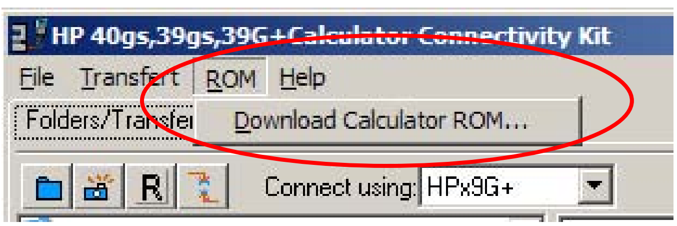
FLASH ROM

Unlike all their predecessors, the hp 39gs & hp 40gs contain *flash ROM*. A ROM chip contains "Read Only Memory" and is used to contain data which must be preserved even when the batteries are removed. For this reason a ROM chip is always used to contain the operating system for the calculator. Data that you create, on the other hand, is stored in RAM ("Random Access Memory") and this memory lasts only as long as power is supplied to it. RAM data is lost when you remove the batteries or do a complete reset.

In normal ROM the contents of the chip are burned in at the factory and can't be altered. The difference with the hp 39gs & hp 40gs is that the ROM used is a 'flash' ROM. This is a special chip where, although the contents are preserved when the batteries are removed or a reset is performed, the contents are not frozen permanently. Clearly this is far more useful than normal ROM but it is only recently that the price of flash ROM has come down to the point where it can be used in calculators.

The advantage of flash ROM is that it means that the operating system of the calculator can be updated if it turns out to contain bugs that HP didn't spot before it was released, or if HP wants to add new features to the operating system. The hp 39gs & hp 40gs still contain a small amount of traditional ROM that can't be altered, but this contains little more than information the calculator needs on how to reboot itself and how to load a new operating system into the flash ROM when it becomes available.

Any updates to the calculator's ROM can always be found on HP's website at http://www.hp.com/calculators and information on them will also be found at the author's website The HP Home view at http://www.hphomeview.com. At the time when this text was written no updates had been released.



The process of performing an update is somewhat similar to that of downloading an aplet. It is accomplished using the Connectivity software (see right).

Full instructions will be given in the package that you will download from HP but the process is outlined briefly below. One stage requires three hands and it is best to have a friend to help!

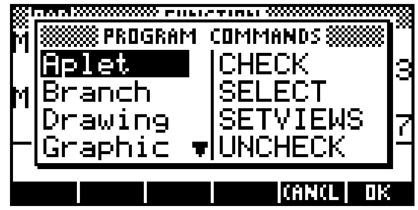
- Choose the file containing the new ROM using the menu option on the Connectivity software.
- Disconnect the cable from the calculator.
- While holding down both the [+] and [−] buttons, perform a reset on the calculator by gently inserting an unbent paper clip into the hole on the back of the calculator in order to press the button inside. This may require two people to do.
- From the menu that appears, choose option 1 to download a new ROM
- From the next menu, choose option 1 to download via USB
- Click on the OK button on the PC to initiate the download.

The process is not quick but requires at the very most half an hour.

Generally any user memory will be lost as part of the updating process. Even if it is not, the instructions that come with the update will almost certainly require that you perform a full reset after the update. Failing to do so might cause the calculator to lock up, requiring a reset anyway. Consequently it will be necessary to save all your aplets, programs and notes to a PC before the update.

A concern to educators may be whether an hp 39gs user can load the operating system of an hp 40gs and thereby gain access to a CAS. This is not possible because the hp 39gs has less flash ROM installed than the hp 40gs. The operating system of an hp 40gs is quite a bit larger than that of the hp 39gs due to the inclusion of the CAS. It will not fit onto the ROM chip of an hp 39gs and any attempt to try will result in an expensive paperweight. At best it would be necessary to find a valid ROM file for an hp 39gs and download that to take the place of the incorrect and partial download done.

PROGRAMMING COMMANDS



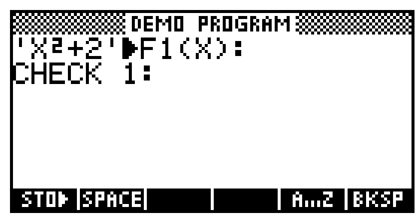
As was explained in a previous chapter, the hp 39gs and hp 40gs are supplied with a simple and easy to use programming language called HP Basic.

All programming commands can by typed in by hand but, as with the **MATH** commands, can also be obtained from a menu. Press **SHIFT CMDS** to display this.

During the course of using the calculators in the classroom and creating hundreds of aplets and programs for them I have found that there are certain commands which are used regularly. In this section I will only be covering those commands which I have used regularly and so regard as important. These may not be the same as the ones you regard as important. If so, consult the manual.

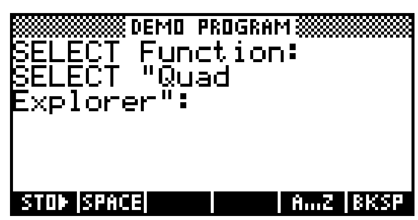
The Aplet commands

CHECK n, UNCHECK n



These commands put or remove a check next to the equation whose number is given by *n*. An interesting bug is actually quite useful: if you **UNCHECK 0** then all equations are unchecked instead of only equation 0. Unfortunately the same is not true for the **CHECK** command. As they say in the trade: "It's not a bug, it's a feature!".

SELECT <name>



This is used to set the active aplet if necessary. If the name has spaces in it then it must be enclosed in quotes. This is not usually required as the program will normally be called by the active aplet anyway.

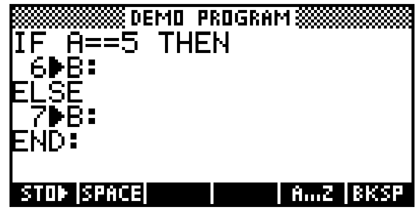
I have only used it with 'stand-alone' programs not attached to an aplet so that they can temporarily 'borrow' abilities belonging to an aplet. However, it could also be used to create an aplet that had two 'parents' if you required it to inherit abilities from both. You could then swap from one parent to the other using this command. This could be quite cumbersome but might add some powerful features.

SETVIEWS <prompt>;<program>;<view number>

This absolutely critical command is covered in great detail on page 259.

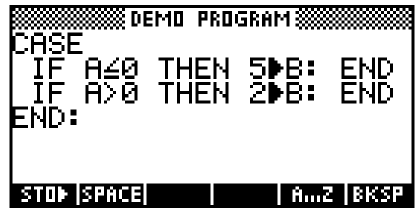
The Branch commands

IF <test> THEN <true clause> [ELSE <false clause>] END



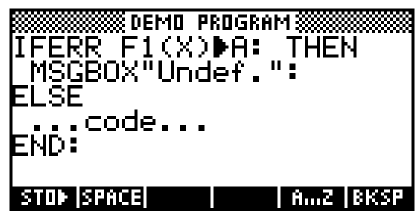
Note the need for a double **=** sign when comparing equalities. Any number of statements can be placed in the true and false sections. Enclosing brackets are not required, as they are in some other languages.

CASE <if clauses> …END:



This command removes the need for nested **IF** commands but is only worth it if you have more than two or three nested **IF**s. Note that colons are not required for the **END**s which terminate the internal **IF** clauses.

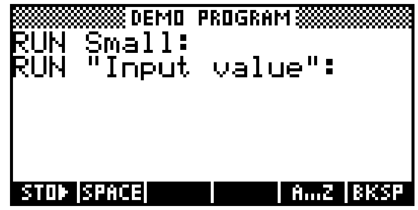
IFFERR <statements> THEN <statements> [ELSE <statements>] END



This can be used to error trap programs where there is a possibility of something going wrong which would normally crash the program, such as evaluating a function at a point for which it is undefined. By trapping the suspect code you can supply an alternative which will perform some other action. This will tend to make your programs more user friendly and is a very good idea!

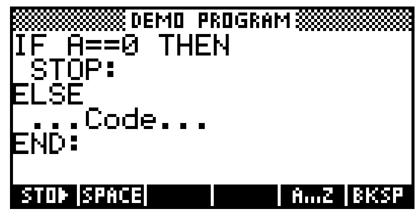
RUN <program name>

This command runs the program named, with execution resuming in the calling program afterwards. If a particular piece of code is used repeatedly then this can be used to reduce memory use by placing the code in a separate program and calling it from different locations. See the **SETVIEWS** command for information on how to link a program to an aplet when it does not appear on the primary menu. Note that if the name has spaces in it then it must be enclosed in quotes.



STOP

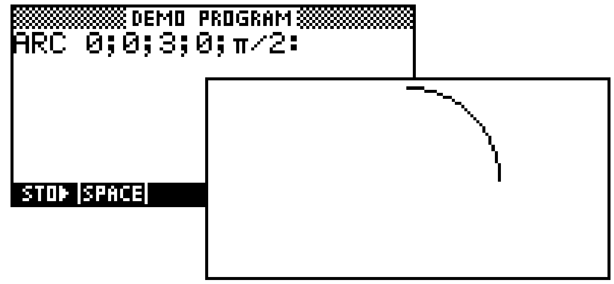
This command can be used to abort execution of a program. Control resumes in the **HOME** view.



The Drawing commands

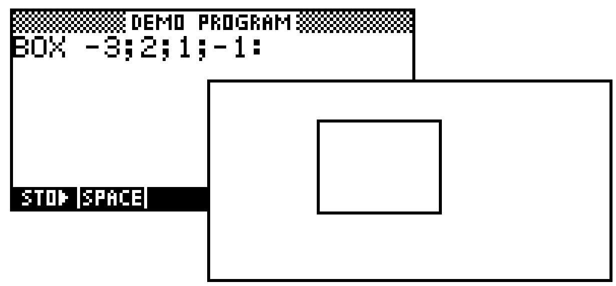
ARC <x-center>;<y-center>;<radius>;<start angle>;<end angle>

This command draws an arc on the screen. It uses the current values in the **PLOT SETUP** view as the screen coordinates and the settings in the the **MODES** view for angle format. This command is unfortunately quite slow.



BOX <x_1>;<y_1>;<x_2>;<y_2>

This draws a rectangular box on the screen using (x_1,y_1) and (x_2,y_2) as the corners. The coordinates are relative to the settings in the **PLOT SETUP** view.



ERASE

This command erases the current display screen.

FREEZE

This command halts execution until the user presses any key. Execution resumes on the next line of the program.

LINE <x_1>;<y_1>;<x_2>;<y_2>

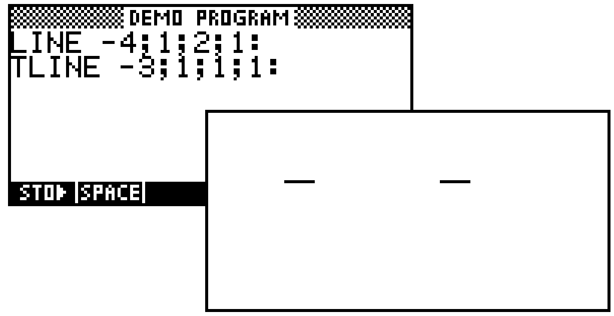
This draws a line on the screen using (x_1,y_1) and (x_2,y_2) as the ends. The coordinates are relative to the current axis settings in the **PLOT SETUP** view.

PIXON <x>;<y> and PIXOFF <x>;<y>

This command turns a pixel point on or off at the specified point. The coordinates are relative to the current axis settings in the **PLOT SETUP** view.

TLINE <x_1>;<y_1>;<x_2>;<y_2>

This command is the same as **LINE** except that the line drawn reverses the current set/unset value of all pixels. This means that it can be used to erase previously drawn lines.



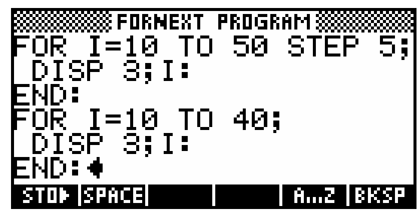
If you would like to see this command in action, download the aplet called “Sine Define” from the author’s website *The HP HOME view* (at http://www.hphomeview.com). This aplet contains extensive use of this command.

The Graphics commands

See the chapter "Programming the hp 39gs & hp 40gs" beginning on page 255 for examples illustrating some of the graphics commands that I have used regularly. Consult the manual for more.

The Loop commands

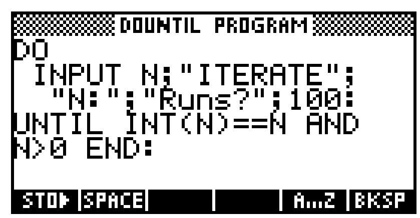
FOR <variable> = <start value> TO
<end value> [STEP <increment>] <statements> END



For those familiar with the Basic language in other forms, this is a standard FOR…NEXT command, except without the 'NEXT'. The **STEP** value is optional and is assumed to be 1 if not stated.

Note: Whatever you do, don't use NEXT to terminate the loop! It doesn't register as an error but all sort of strange things happen!

DO <statements> UNTIL <test clause> END



This loop executes the statements within it until the test clause evaluates as true. It must execute at least once because the test is not done until the end of the loop.

The example right checks for a positive integer from the **INPUT** statement. To be even more user friendly you could let the user know what they had done wrong by adding another few lines of code within the **DO** loop of..

```
IF INT(N)≠N OR N≤0 THEN
  MSGBOX "Enter a positive integer only":
END:
```

WHILE <test clause> REPEAT <statements> END

This is similar to the **DO…UNTIL** loop except that the test clause is evaluated before starting so that the loop may not be executed at all.

These two loops are very similar and, except in some odd situations, are interchangeable.

BREAK

This command will exit from the current loop, resuming execution after the end of it.

> **Calculator Tip**
> There is no GOTO <label> command in the language as there is on some other calculators. This is often seen as a disadvantage by inexperienced programmers but in fact it usually results in code which is far more easy to read and much less likely to have bugs in it. Labels generally result in what is known in the trade as 'spaghetti code', so called because it has odd GOTO's that lead all over the place, intertwining and making it almost impossible to follow. There is no situation in which a GOTO statement cannot be replaced by one of the loops or conditional branches available in the HP Basic language.

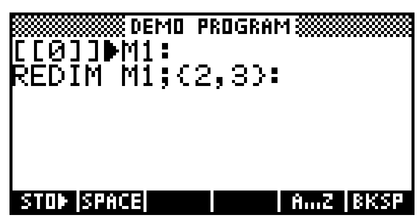
The Matrix commands

EDITMAT <matrix var>

This command pops up a window in which the user can edit or input a matrix with an **OK** key at the bottom. When the user presses **OK**, execution resumes after the **EDITMAT** statement.

REDIM <matrix var>;<size>

This command is very useful if the size of a matrix is not known in advance. The user might be prompted to input the size and then these values used to resize it. Note that the dimensions must be supplied as a list variable. The **SIZE** command can also be used in this context as it returns a list variable when used with a matrix. For example, you could find the current **SIZE** of a matrix and then use **REDIM** to add a new row or column.



The Print commands

These commands were supplied for use with the battery operated HP infra-red thermal printer that is designed for use with any of the 39g family. This printer was originally designed for the first calculator in this family, the hp 38g, released in 1995. Very few of them sold because it was far easier to simply use the connectivity software to capture screens and images and then just paste them into a document on the PC. At the time of writing I was unable to find any reference to it on the HP website and it may no longer be available.

Because it uses the infra-red port the commands will not operate with the hp 40gs. There may be a brief display of the communications symbol, a small arrow, in the top right corner of the display but there will be no other result.

PRDISPLAY

If you place this command in a program then the current display will be sent to the infra-red printer.

PRHISTORY

This command, whether issued in the **HOME** view or in a program, will send the entire contents of the History to the infra-red printer.

PRVAR <variable>

This command, whether issued in the **HOME** view or in a program, will send the value of the variable to the infra-red printer. This can be used to capture and send images of graphs without need for programming as follows:

- set up the graph or image as required
- press **ON+PLOT** to capture the image and store it in grob **G0**
- in the **HOME** view, issue the command **PRVAR G0**.

The Prompt commands

BEEP <frequency>;<duration>

This will use the piezo crystal in the calculator to create a sound of the specified frequency for the specified duration (in seconds). The resulting frequency is not terribly accurate, varying by up to 5% from one calculator to the next and depending also on the temperature. If you want accuracy then use a piano! The volume is also not very loud because of concerns with interruptions to tests and examinations. Any frequency can be produced using the **BEEP** command but most people are interested in using it to play tunes. The information below will help with this.

The frequencies of the twelve semi-tone jumps in the normal Western harmonic scale form a geometric sequence, and since the ratio from C to C' is 2, the ratio for each semi-tone must be $\sqrt[12]{2}$. The standard frequency used in tuning instruments is usually 440 cycle/sec for the note A. Since much of the simple music used by students is written using the scale of C, we use $440/(\sqrt[12]{2})^9$ to find the frequency of C as 261.6 cycles/sec.

```
.5▶T:
261.6▶C:
12 NTHROOT 2▶R:
R²C▶D:
R²D▶E:
R*E▶F:
R²F▶G:
R²G▶A:
R²A▶B:
```

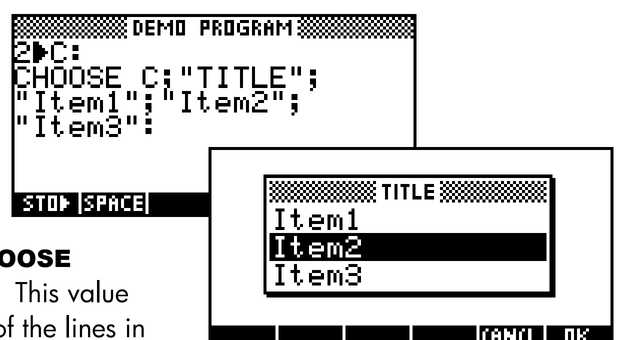
```
BEEP F;T/2:
BEEP G;T/2:
WAIT .01T:
BEEP G;T/2:
BEEP F;T/2:
BEEP G;2.5T:
BEEP F;T/2:
BEEP G;T/2:
BEEP A;T/2:
BEEP G;T/2:
BEEP F;1.5T:
```

We can use this to form a standard 'header' for any program we want to use to play music. The header shown right in the rounded box sets up the scale of C major. The code which then follows plays the first two bars of the tune "Strangers in the Night".

In this header, the duration of a note (T) is set to 0.5 seconds. It is easy to change the tempo of the music by adjusting this. In this case you may find that the music sounds a little better with T set to 0.55 or 0.6 seconds. T is a crotchet, T/2 a quaver etc.

CHOOSE <variable>;<title>;<menu option1>;….

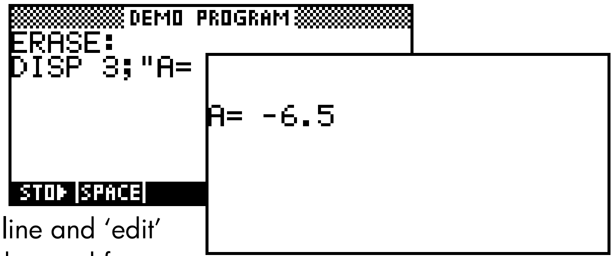
This command pops up a menu with the title specified and with however many options follow. The number of the menu option highlighted when the user presses **OK** is returned in the variable.



It is particularly important to the use of this command to realise that the initial value of the variable before the **CHOOSE** statement determines which option is initially highlighted. This value must be a valid one in the sense that it must refer to one of the lines in the list you have defined. Assigning an initial value outside the range of the menu may crash the program. In particular, the default value for any variable is zero and this is not a legal list line so failing to pre-define a value may crash your program. If the user presses **CANCL** then a value of zero is returned but the program will still continue to execute from that point unless you include code to check for the cancellation and take action to terminate it or take some other action.

DISP <line number>;<expression>

This command breaks the display up into 7 lines and allows output to them.

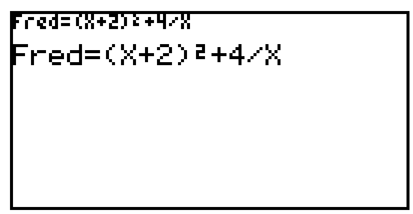


Using the **DISP** command on a line wipes that entire line to the right hand end of the screen before display. This means that it is not possible to write over an existing line and 'edit' material already present. The **DISPXY** command should be used for this purpose but is a little more complex to use.

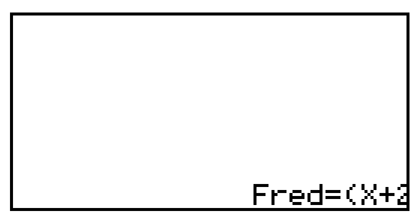
DISPXY <xpos>;<ypos>;<font>;<expression>

This command displays the text/object/result contained in <expression> at the screen position specified using the font specified. An extensive example can be found in the chapter "Programming the hp 39g & hp 40gs" on page 274.

The xpos and ypos values are positions which are relative to the values of **Xmin**, **Xmax**, **Ymin** & **Ymax** as set in the **PLOT SETUP** view. The font number can be either 1 (small) or 2 (large). For example, the program shown left will produce the result shown below right if the axes are set to the default values of -6.5 to 6.5 and -3.1 to 3.2.



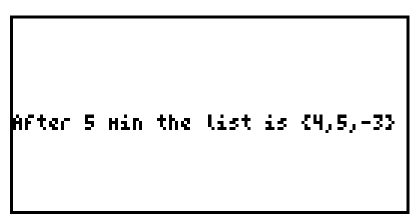
The xpos and ypos values refer to the top left corner of the text when displayed. The clipping rules are a bit complicated, probably due to the fact that the command was done in a bit of a hurry (in a hotel room late one night in Chicago after I'd pleaded with the programmers!).



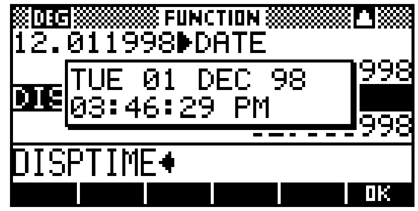
If the xpos, ypos corner is off the top of the screen or off the left edge of the screen then the string will not be displayed even if part of it is onscreen. If there is sufficient room vertically for the string but part of it would display off the right edge of the screen then it will be partially displayed (see right). On the other hand, if we were to move the text one more pixel downwards so that it would display partially off the bottom of the screen then it will not display at all. The best way to satisfy yourself of this is to experiment.

Finally, the value of <expression> can include variables of any kind and can mix text and variables. For example you could have a command:

DISPXY -6.5;0;1;"After "N" min the list is "L0:



DISPTIME

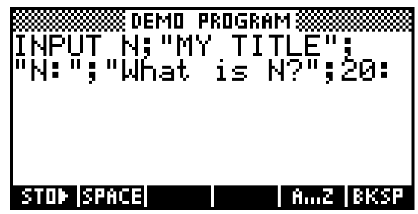


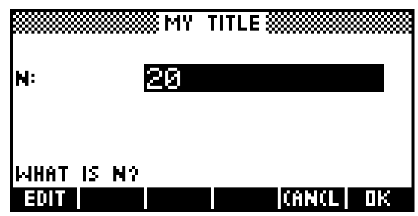
This command pops up a box displaying the calculator's internal time and date. These can be set by storing values to the variables **Time** and **Date**. Suppose the current time is 3:46:29 pm on the 1st of December, 1998 then you would store 15.4629 to **Time** and 12.011998 to **Date** as shown in the screen shot right. Having the correct time and date set has no effect on the operation of the calculator!

GETKEY <variable>

This pauses the execution until the user presses a key and stores the code for the key into the variable for later use. See the manual for how the value of the key changes according to whether it is pressed with or without the **SHIFT** or **ALPHA** keys. A key code of 53.1 would mean row 5, column 3 and un-shifted. It can't readily be used for games because execution pauses instead of continuing in the background.

INPUT <variable>;<title>;<prompt>;<message>;<default value>

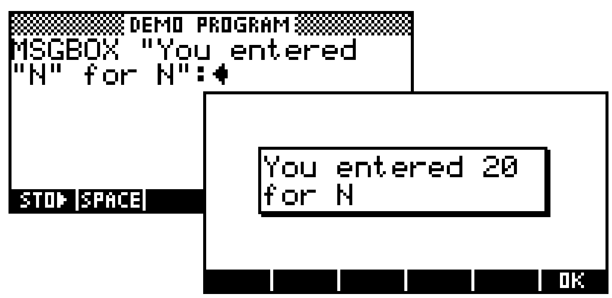




This command puts up an input view which can be used to obtain responses from the user. The degree of control over appearance is quite high as can be seen in the example.

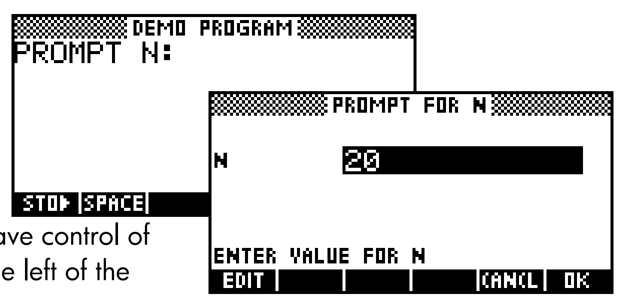
If you want the default value to be whatever the user last input then use **INPUT N; ....; N** instead. If you do this then you should consider storing an initial reasonable value into **N** before the first use of the **INPUT** command.

MSGBOX <expression>



This puts up a box with the text/expression you specify. If you want a new line started within the box then just enclose a pressing of the **ENTER** key within the quotes.

PROMPT <variable>



This is a short form of the **INPUT** statement for those that don't require such precision of control over appearance. The default value is the current value of the variable. Using the **PROMPT** command you don't have control of the title and prompts at the bottom of the screen and to the left of the input field.

WAIT <duration>

This command pauses execution for the specified number of seconds. Execution resumes at the next statement after the **WAIT** command.

APPENDIX A: SOME WORKED EXAMPLES

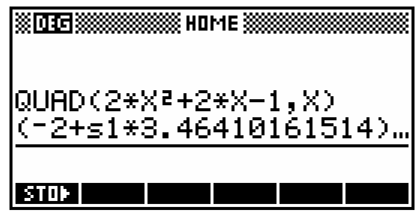
The examples which follow are intended to illustrate the ways in which the calculator can be used to help solve some typical problems. In some cases more than one method is shown. In some cases the method is chosen more to illustrate the capabilities of the calculator than because it is necessarily the most efficient method. Sometimes these problems are quoted elsewhere in the book and repeated here for convenience.

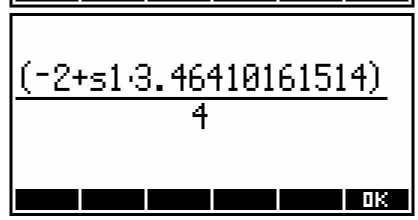
Finding the intercepts of a quadratic

Find the x intercepts of the quadratic equation $g(x) = 2x^2 + 2x - 1$

Method 1 - Using the QUAD function in HOME.

This method is shown right, using the **SHOW** key in the bottom view. This is probably not a method that one would use in general but it has the slight advantage that the answer is given in the same form that you would see it if you used the Quadratic formula. Just **COPY** the result, edit and square the decimal part to find the value of the discriminant if you need the result in surd format. The '**S1**' is the calculator's version of the ± sign. Just **COPY** the result and remove the S1 to obtain the positive solution, replacing the + with a - to obtain the other. This method is only of use if the question said "Show working" because it doesn't give the answer directly.



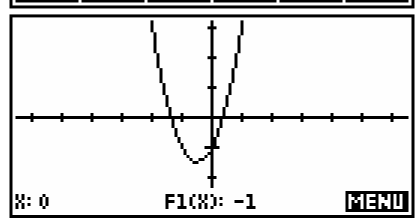


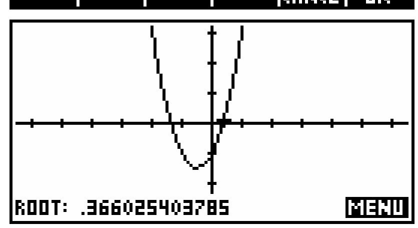
Method 2 - Using the Function aplet.

Shown right. Enter the function into the **SYMB** view, use the **VIEWS** key and choose '**Decimal**'. If the axes don't suit, then use the **ZOOM** options. Now use the **FCN** option of *Root* to find the two roots. One result is shown.

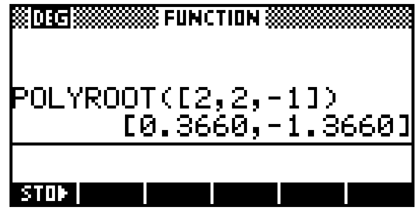
This is clearly the best method and has the advantage that you can see the graph clearly.

Note: Using the **FCN** *Root* method does not depend on the graph being on the screen. The algorithm will still find roots even if they are not currently visible.





Method 3 - Using the **POLYROOT** *function*

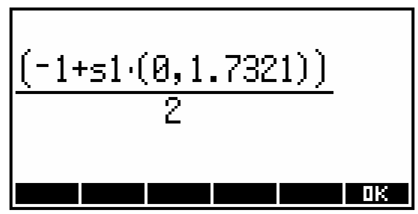


The advantage of this is that it can be done in the **HOME** view and is quick and easy. It also has the advantage that it returns complex roots as well. See page 84 for a method of copying the results to a matrix so as to gain easier access to them. This method is highly recommended for polynomials in general.

Finding complex solutions to a complex equation

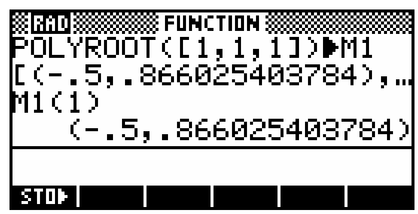
Find the roots of the complex equation $f(z) = z^2 + z + 1$.

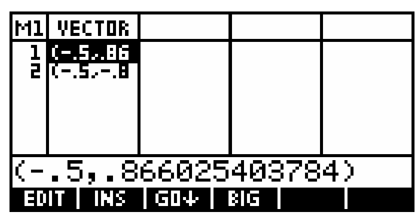
Method 1 - Using the QUAD function



Since this is a quadratic it can be done with the **QUAD** formula mentioned in example 1., since it is capable of giving complex results. This is shown right, rounded to 4 dec. pts. It's up to you of course to realize that (0,1.7321) is $\sqrt{3}\,i$ but if you don't recognize it then **COPY** just that portion and square it. The '**S1**' means ± . The only real advantage of this method is that it gives the answer in the same format as the quadratic formula and this may be of use.

Method 2 - Using POLYROOT





An alternative method is to use the **POLYROOT** function and store the results to a matrix. This offers the advantage of being able to examine the result more easily by **EDIT**ing the matrix, and also of being able to access each root by referring to the matrix elements in a calculation (eg M1(1), M1(2) etc.). See page 84 for more details on this.

The disadvantage of this method is that for higher degree polynomials with a mixture of real and complex roots, all roots are shown in complex form. This can make the real roots harder to isolate and use. However, it is still the best method.

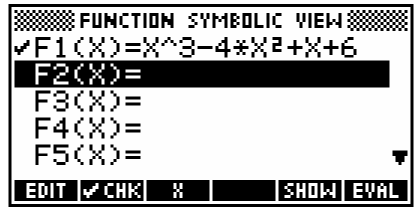
Method 3 - Using the CAS on the hp 40gs

See page 309 for an example of finding roots of real and complex polynomials using the CAS on the hp 40gs.

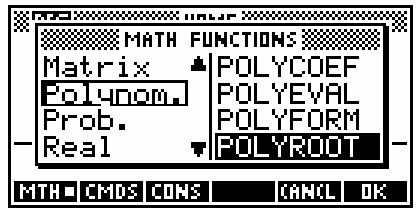
Finding critical points and graphing a polynomial

For the function $f(x) = x^3 - 4x^2 + x + 6$ …

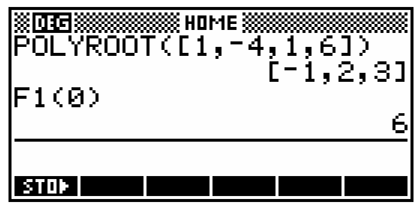
(i) find the intercepts.
(ii) find the turning points.
(iii) draw a sketch graph showing this information.
(iv) find the area under the curve between the two turning points.



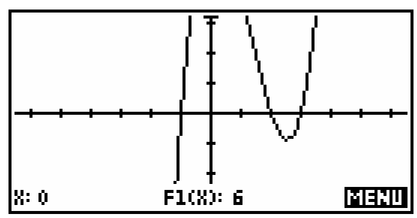
Step 1. Enter the function into the **SYMB** view of the Function aplet, so it is available for plotting.



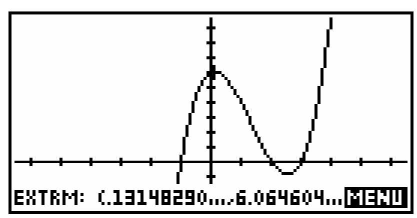
Step 2. Use the **POLYROOT** function to find the roots. This function is in the **MATH** menu in the *Polynom.* group. See page 204 for more information.



The results show that the x intercepts are $(-1,0),(2,0)$ and $(3,0)$. The y intercept is found by evaluating **F1(0)** in the **HOME** view giving the point (0,6).

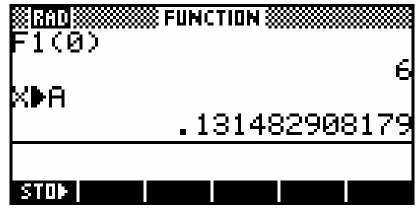


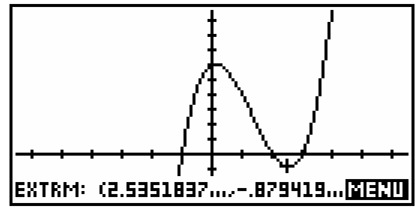
Step 3. Switching to the **PLOT** view via **VIEWS** - **Decimal**, you will find that the function does not display as well as it could. Since it is the y axis that is not displaying enough, we will use the '**Y-Zoom Out**' option in the **ZOOM** menu after first setting the **Zoom factors** to 2 rather than 4 (which is too drastic). The **Zoom factors** setting is also found in the **ZOOM** menu.



Now use the **FCN** pop-up menu to find the *Extremum* (both left and right). The snapshot right shows the left-hand turning point of (0.131,6.065).

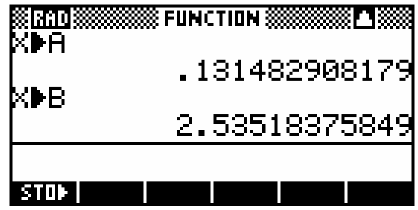
Step4. Because I know that part (iv) of the question requires me to re-use these extremum values in an integration (which I would like to be as accurate as possible), I am going to 'save' the extremum value just found. I change into the **HOME** view and store it as shown in memory **A**.



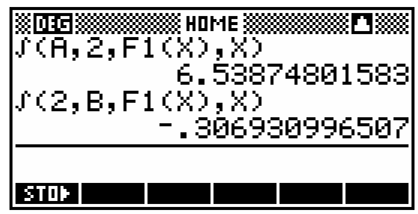


Note: You MUST store the point of interest before moving the cursor in the **PLOT** view. As soon as the cursor moves its new position over-writes the extremum value. If you want the y coordinate, just evaluate **F(X)**.

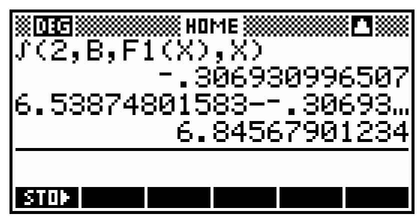
Now use the **FCN** *Extremum* tool again to find and store the x coordinate of the second turning point into memory **B** as shown.



Step 5. The **PLOT** view shows that part of the area we require for part (iv) is negative, so we need to know the x intercept between the two turning points. Fortunately we know from Step 2 that it is the point (2,0). If we did not know this already, then we could use the **FCN** menu again, retrieving this time from the **VAR** menu the variable called '**Root**' and perhaps storing that into memory **C**.



We will evaluate the integral in the **HOME** view where you can use the accurate values you stored in Step 4. It needs to be done in two parts and added (subtracted actually to reverse the sign of the negative part). This is shown right, with the **COPY** key having been used to add the values.

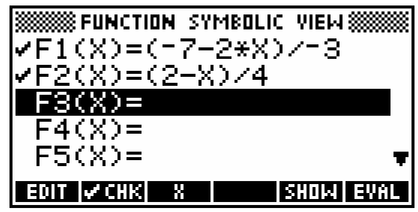


Solving simultaneous equations.

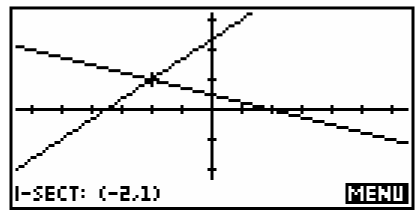
Solve the systems of equations below:

(i) $\left.\begin{aligned} 2x-3y&=-7 \\ x+4y&=2 \end{aligned}\right\}$ (ii) $\begin{aligned} 2x-y&=4 \\ -3x+2y-z&=-10.5 \\ x-3y+z&=10.5 \end{aligned}$

Method 1 - Graphing the lines



Because the first set of equations is a 2x2 system it can be graphed in the Function aplet. To do this it is necessary to re-arrange the functions into the form y = …… and store them into **F1(X)** and **F2(X)** in the **SYMB** view of the Function aplet. Switch to the **PLOT** view and use the **FCN** **Intersection** tool to find the point of intersection.



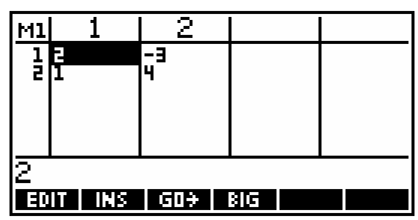
It is worth noting that although the point of intersection is on the screen here, this is not necessary. The **FCN** **Intersection** tool will work even if neither line is visible on the currently set axes.

Method 2 - Using a matrix

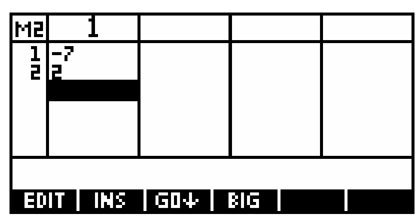
Step 1. Rewrite $\left.\begin{aligned} 2x-3y&=-7 \\ x+4y&=2 \end{aligned}\right\}$ as $\begin{bmatrix} 2 & -3 \\ 1 & 4 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}=\begin{bmatrix} -7 \\ 2 \end{bmatrix}$

This means that $\begin{bmatrix} x \\ y \end{bmatrix}=\begin{bmatrix} 2 & -3 \\ 1 & 4 \end{bmatrix}^{-1}\times\begin{bmatrix} -7 \\ 2 \end{bmatrix}$

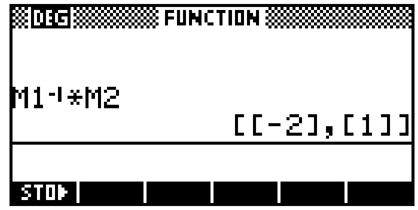
Step 2. Switch into the Matrix Catalog (**SHIFT MATRIX**). Position the highlight on **M1** and press **EDIT**. Enter the matrix shown right.

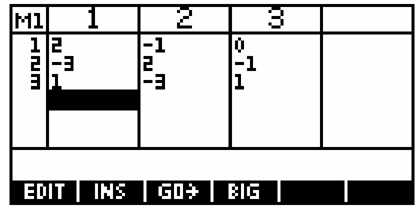


Press **SHIFT MATRIX** to change back to the catalog view and create **M2** as shown below right. Note that setting **GO↓** will make it easier to enter **M2**.

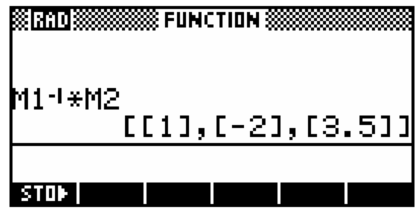


Step 3. Change into the **HOME** view and enter the calculation **$M1^{-1}*M2$**. The result is the (x,y) coordinate of the solution displayed as a matrix.



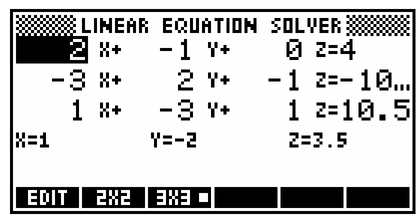


A similar method can be used to solve the second 3x3 system of equations. The matrix **M1** and the result are shown right.



Method 3 - Using the Linear Solver aplet

This method uses an aplet called the **Linear Solver** which was added into the new hp 39gs and hp 40gs. For earlier models there is a similar aplet available from the internet called the “Simult 3x3” aplet.



It allows easy solution of 2x2 and 3x3 systems of linear equations in a format which is more user friendly than the use of matrices for student who are not familiar with them. The disadvantage is that it shows no working.

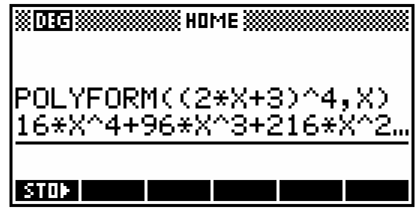
Note: If your simultaneous linear equations have algebraic coefficients then you will not be able to use any of the above methods because they will all substitute the current value for the coefficients rather than assuming they are symbolic. If you are fortunate enough to own an hp 40gs rather than an hp 39gs then you can use the CAS for this. See page 346 for an example.

Expanding polynomials

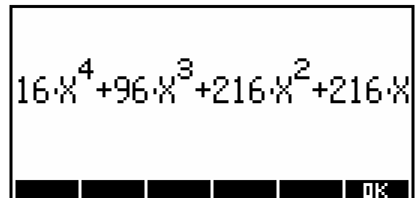
Expand the expressions below.

(i) $(2x+3)^4$

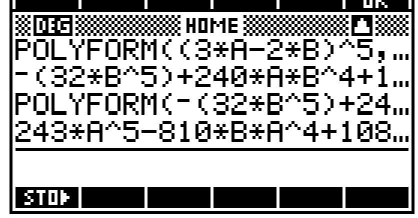
(ii) $(3a-2b)^5$



(i) Use **POLYFORM((2X+3)^4,X)** to expand the polynomial. Use the SHOW key to examine the result.



Result: $16x^4+96x^3+216x^2+216x+81$



(ii) Use **POLYFORM((3A-2B)^5,B)** to expand the polynomial as a function of **B**. Then use the polynomial function again, COPYing the result from the first expansion and expanding this time as a function of **A**. The SHOW key can then be used to view it, using the left and right arrows to scroll the screen left and right.

Result: $243a^5-810a^4b+1080a^3b^2-720a^2b^3+240ab^4-32b^5$

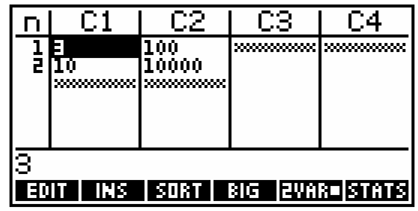
Exponential growth

A population of bacteria is known to follow a growth pattern governed by the equation $N = N_0 \;^{kt} \quad ; t \geq 0$. It is observed that at t = 3 hours, there are 100 colonies of bacteria and that at t = 10 hours there are 10,000 colonies.

(i) Find the values of N_0 and of *k*.

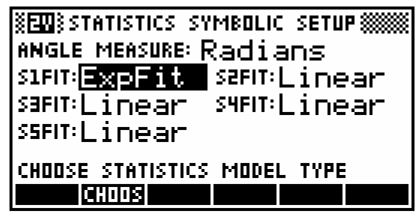
(ii) Predict the number of bacteria colonies after 15 hours.

(iii) How long does it take for the number of colonies to double?

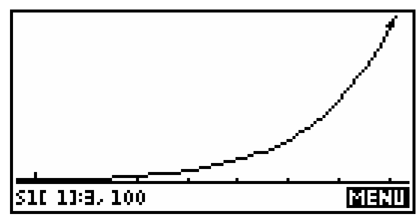


(i) Find N_0 and *k* using the calculator.

Step 1. Start up the Statistics aplet, set it to 2VAR and enter the data given. Change to the **SYMB SETUP** view and specify an *Exponential* line of best fit for the data.



Step 2. Press **VIEWS** and choose *Auto Scale*. If the line shown does not appear then press MENU and then FIT. Wait while the line draws.

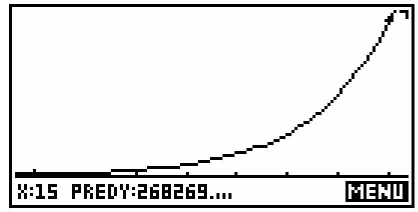


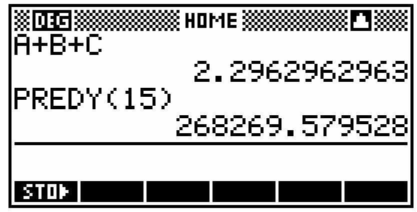
Step 3. Change to the **SYMB** view, move the highlight to the equation of the regression line and press SHOW. Rounded to 4 dec. places, this gives an equation of:

$$N = 13{\cdot}8950e^{0{\cdot}6579t}$$

(ii) Predict N for t = 15 hours.

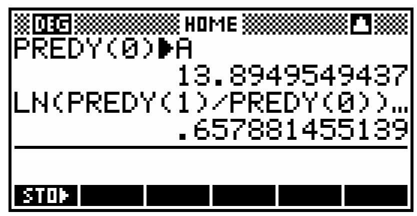
In the **PLOT** view, press up arrow to move the cursor onto the curve of best fit. Now press **GOTO** and enter the value 15. The cursor will jump to the predicted value for x=15, which is currently off screen.





Alternatively, change to the **HOME** view and use the **PREDY** function.

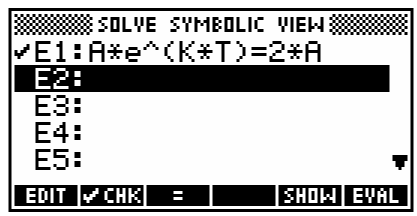
Result: 268 269 colonies.

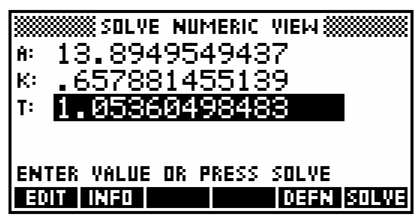


(iii) Find *t* so that $N = \frac{1}{2} N_0$.

Step 1. Find the values of N_0 and *k* and store N_0 into memory A and *k* into memory K, so that it is un-necessary to re-type them.

See page 135 for instructions on finding the parameters from the exponential fit curve.





Step 2. Switch to the Solve aplet and enter the equation to be solved. Changing into the **NUM** view, you should find the values of **A** and **K** already defined, so move the highlight to **T** and press **SOLVE**.

Result: Doubling time is 1.0536 hours.

Solution of matrix equations

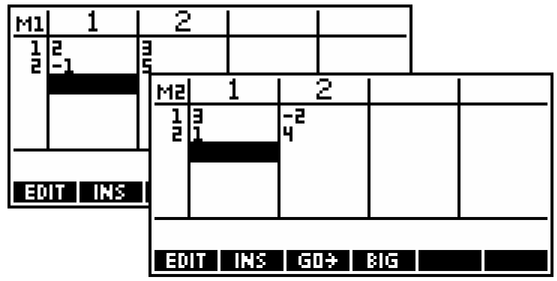
Solve for the value of X in $A(I-2X)=B$

where $A=\begin{pmatrix}2 & 3\\ -1 & 5\end{pmatrix}$, $B=\begin{pmatrix}3 & -2\\ 1 & 4\end{pmatrix}$

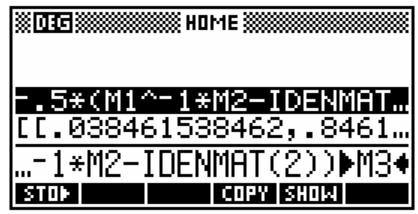
$$\begin{aligned} A(1-2X) &= B\\ 1-2X &= A^{-1}B\\ -2X &= A^{-1}B-I\\ X &= \frac{-1}{2}\left(A^{-1}B-I\right)\end{aligned}$$

The algebraic calculation for this process is shown above right. Having done this, we will now see how to calculate this result on the calculator.

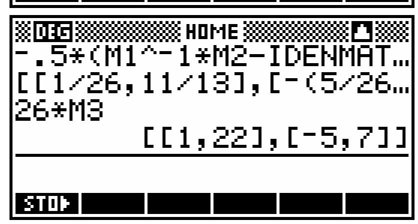
Store the values of A and B into **M1** and **M2** respectively.



Finally use the **HOME** view to calculate the answer, using the function **IDENMAT(2)** to produce a 2x2 identity matrix, and making sure to store the result into **M3**.



In this case the result is a horrible decimal. The fractional equivalent can be found by setting the **MODES** view to *Decimal 6* and then re-evaluating, as shown right. From this it can be seen that the common denominator seems to be 26, so multiplying **M3** by 26 will give a final result of:



$$X=\frac{1}{26}\begin{pmatrix}1 & 22\\ -5 & 7\end{pmatrix}$$

Finding complex roots

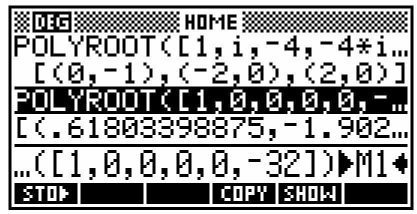
i. Find all roots of the complex polynomial $f(z) = z^3 + iz^2 - 4z - 4i$.

ii. Find the complex roots of $z^5 = 32$.

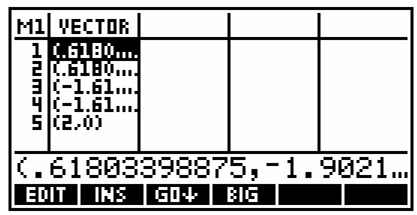
The best way to do this is using **POLYROOT**. I usually **STO▶** the results into a matrix, since the matrices on the hp 39gs and hp 40gs can be complex vectors, not just real valued matrices.

(i) The coefficients can be entered into **POLYROOT** in the form *a+bi* or as *(a,b)*. In this case the roots are integers so there is no need to store it into a matrix.

Coefficients must be in square brackets separated by commas.



(iii) The method is to solve the complex polynomial $z^5 - 32 = 0$, setting the other coefficients to zeros. This is shown in the second **POLYROOT** calculation in the screen shot right.



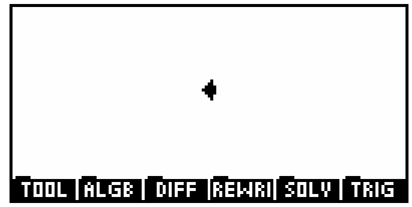
In this case the results are unlikely to be integers so we store them into **M1**. The result is shown below and right. The edit line shows the highlighted element to a greater degree of accuracy. Unfortunately there is no way on the hp 39gs to get exact surds as your answer. As you'll see on the next page, the hp 40gs is more able.

Complex Roots on the hp 40gs

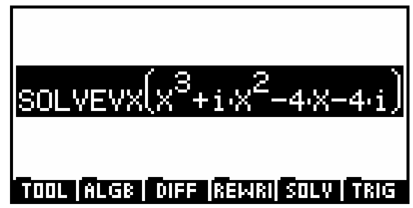
i. Find all roots of the complex polynomial $f(z) = z^3 + iz^2 - 4z - 4i$.

ii. Find the complex roots of $z^5 = 32$.

On the hp 40gs you can obtain exact roots for polynomials using the CAS function **SOLVEVX**. The instructions following assume that the CAS is in its default configuration. See page 324 for more details on the CAS.

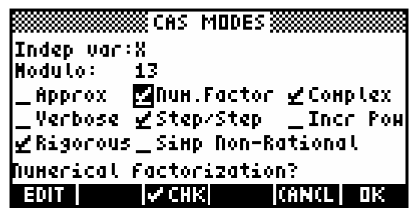


In the **HOME** view, press the **CAS** button to enter the CAS. Press the **SOLV** button to access the Solve menu. Scroll down to find **SOLVEVX** and press **ENTER**.

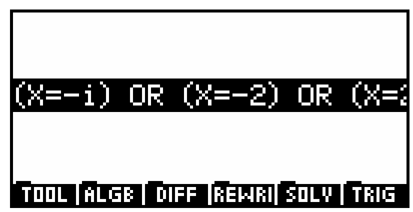


Type in the function using the default variable **X** rather than **Z** and then highlight the entire expression. The result should be as shown right.

If you press **ENTER** at this moment then the result will probably be "**NO_SOLUTION**" because we need to set one of the configuration flags. Press **SHIFT MODES** to enter the configuration menu shown right.

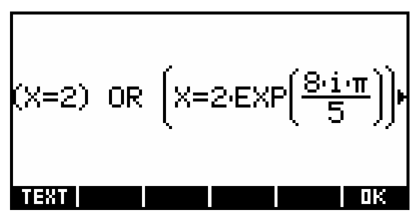


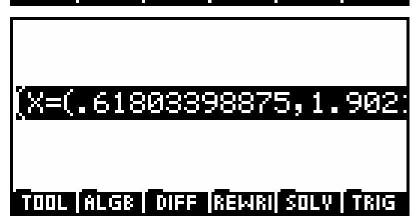
In the configuration screen, move the highlight to the "Num. factor" option shown right and press **✓CHK** to set it. Then press **OK** to quit from the configuration menu and return to the main screen.



Now press **ENTER** to tell the calculator to execute the **SOLVEVX** command. The calculator will then ask if you want to turn "Complex Mode" on. Tell it **YES**. The final result is shown right. Pressing **VIEWS** will allow you to scroll right and left through the solutions.

A similar result can be found for the second equation. The calculator will ask you if you want to turn "Approximation mode" on. If you respond with **YES** then the result will be as shown right. Responding with **NO** instead will result in the second screen below right. Note that "Approx mode" will be left active and you should choose "Exact mode" through the configuration menu if you don't want decimal results from that point on.

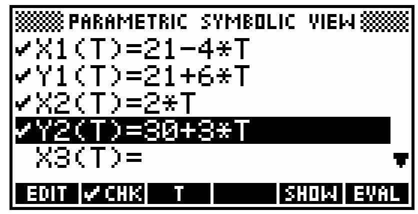




An alternative to using the **SOLVEVX** function is to use **FACTOR**. The difference will be that the results will be in factor form rather than roots.

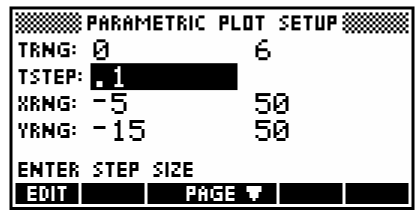
Analyzing vector motion and collisions

*Ship A is currently at position vector 21**i** + 21**j** km and is currently traveling at a velocity of -4**i** + 6**j** km/hr. Ship B is at 30j and traveling at 2i + 3j km/hr. If the ships continue on their present courses, show that they will not collide and find the distance between them at the time of their closest approach.*

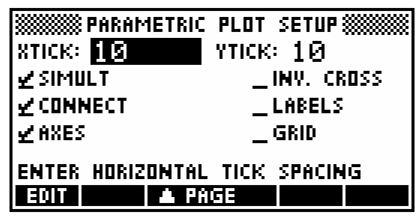


The advantage of the approach shown here is that it is very visual. Obviously there are other methods based purely on calculations.

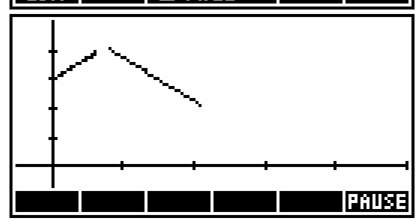
Firstly enter the equations for the ships' paths into the **Parametric** aplet using the first equation pair for ship A and the second for ship B.



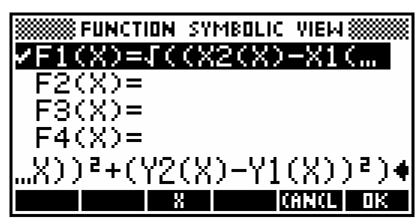
Making a guess at the ships' behavior and position, I will set up the axes as shown on the right. I am assuming that the collision will occur in the first 6 seconds and in the axes range chosen. This can always be adjusted if my guess is wrong.



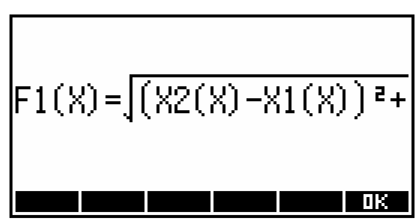
The reason for choosing -15 on the **YRng** is to ensure that the x axis is visible on the screen. Notice in the second screen that *Simult:* must be checked if the plot is to be a good illustration of the ship's movements.



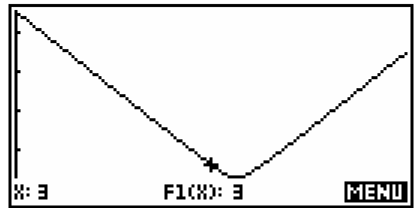
If you've done this correctly then you will see the ship's movements on the plot view. Careful examination of the paths of the ships as they appear on the screen will show visibly that they do not collide. The graph is shown on the right just before closest approach. However we need to verify this.



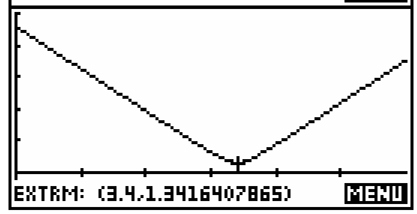
The formula for the distance between the ships is $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ and this can now be entered into the Function aplet as shown right. With an equation this complex it is probably worth checking with the **SHOW** key that you have typed it correctly. Notice that the active variable in Function must be X not the T used in Parametric.



I want to graph this function for the first six seconds but I am not sure what y scale to use so I will set **XRng** to be 0 to 6 in the **PLOT SETUP** view and then choose **VIEWS -** *Auto Scale*. The result is shown right.

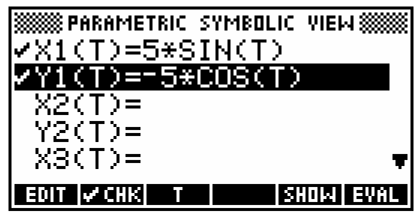


Using **FCN** *Extremum*, I find that the time of closest approach is at *t = 3.4 hours* (3:24 pm) with a separation at that time of *d = 1.3416 km*. The y axis has been adjusted slightly to make the x axis visible.

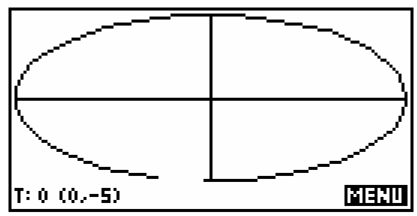


Circular Motion and the Dot Product

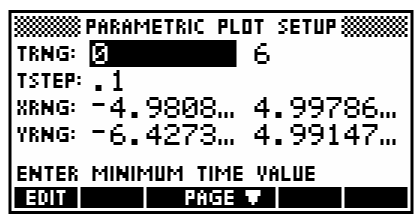
A particle's position at time t seconds is given by $r = 5\sin\theta\, i - 5\cos\theta\, j$ *. Draw a sketch of its path for the first 6 seconds of movement. Show that its path is circular.*



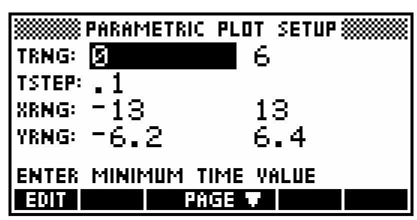
The first step is to graph the particle's path. We go into the Parametric aplet and enter the rule into the first equation **X1**, **Y1**.



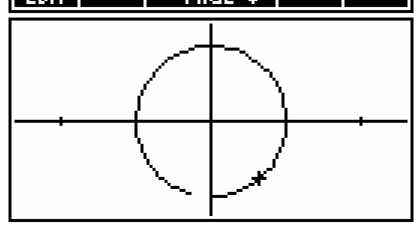
After setting **TRng** to 0 to 6 and then using **VIEWS -** *Auto Scale* we get a result which is does not look like a circle, but this may be due to distortion through not using a 'square' screen.



If we inspect the **PLOT SETUP** view, we can see that the settings are nearly double those of the default 'square' screen of -6.5 to 6.5 and -3.1 to 3.2 In the **PLOT SETUP** view, press **SHIFT CLEAR** to set everything back to default, then double each setting for the x and y axes and set **TRng** back to 0 to 6.



Pressing **PLOT** again will make it clear visually that this is a circle and we can copy this diagram onto our test paper as the required graph. Note the gap at the bottom because the **TRng** stops at 6 instead of 2π.



Of course, this is not really a proof of circularity!

Inference testing using the Chi² test

A teacher wishes to decide, at the 5% level of significance, whether the performance in a problem solving test is independent of the students' year at school. The teacher selected 120 students, 40 from each of Years 8, 9 & 10, and graded their performance in a test as either A or B.

	Grade awarded		
Year	A	B	Total
8	22	18	40
9	26	14	40
10	27	13	40
Total	75	45	120

The table above right shows the results of his testing.

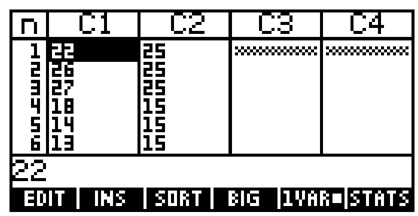
The hypotheses being tested are:

H_0: There is no relationship between grades awarded and years at school. They are independent.
H_A: There is a relationship.

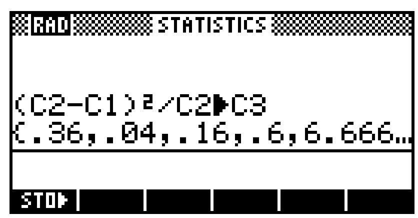
If H_0 is true then the expected frequencies should be those in the table on the right.

	Grade awarded		
Year	A	B	Total
8	25	15	40
9	25	15	40
10	25	15	40
Total	75	45	120

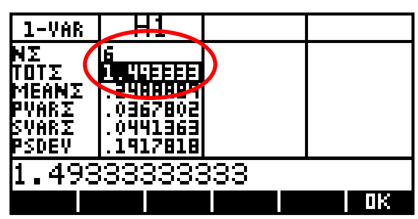
Enter the observed and expected frequencies into columns **C1** and **C2** of the Statistics aplet.



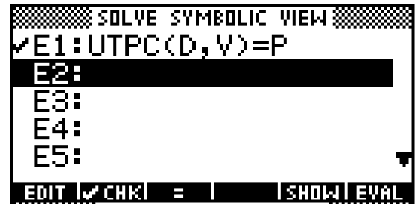
In the **HOME** view, perform the calculation shown right. This calculates the individual χ^2 values ready for summing as per the formula $\chi^2 = \sum \frac{(O-E)^2}{E}$, where O and E are the observed and expected frequencies.



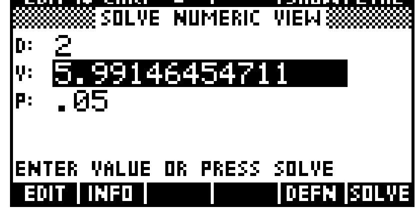
After first changing to the **SYMB** view to register **C3**, the required total for **C3** can be found in the **STATS** view. In this case, 1.493.



Changing into the Solve aplet we can enter a formula which will allow us to calculate values from the *Chi*^{2} distribution using the **UTPC** function.



With a 3x2 contingency table the number of degrees of freedom are 2. To find the critical $\chi^2_{0.05}$ value, we enter values of 2 for **D** (the degrees of freedom) and 0.05 for **P** (the probability) and then move the highlight to **V** (the value) and press **SOLVE**. As it turns out, the required critical value is 5.99 and so we would accept the null hypothesis and conclude no relationship.

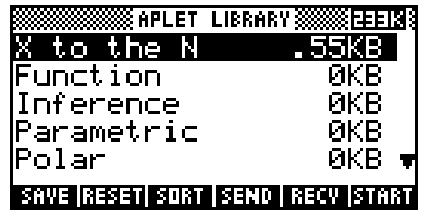


43

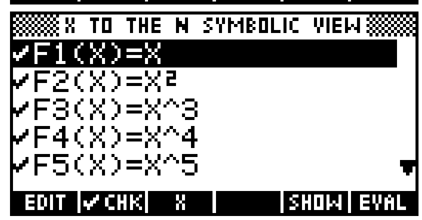
APPENDIX B: TEACHING OR LEARNING CALCULUS

There are many ways that the teaching or learning of functions and calculus concepts can be enhanced with the aid of a graphical calculator. Some of them are listed below:

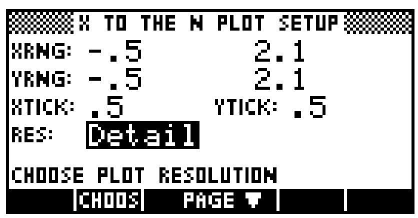
Investigating the graphs of $y=x^n$ for n *an integer*

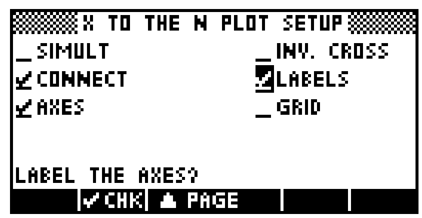


This is a task often given to introductory calculus classes and can be done most economically by setting an investigative task, perhaps for homework.

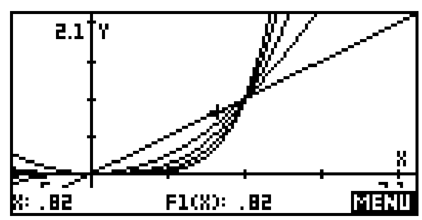


Save a copy of the Function aplet under the name of "X to the N". Saving the aplet will allow you (the teacher) to send it to your students' calculators.





Into this aplet, enter the functions shown and set the axes as shown. The choice of x axis here means that each pixel will be 0.02 apart, which means that when students trace the cursor along the graph they will obtain 'nice' values.

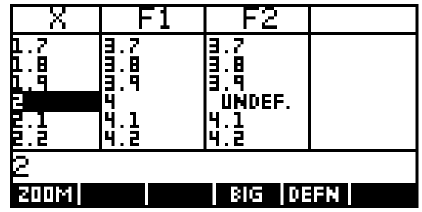


This aplet can now be sent to each student's calculator at the end of a lesson using the infra-red link. Accompanying questions should address the issues below, and students should be required to either hand in a short written response, or contribute to a verbal discussion the next day.

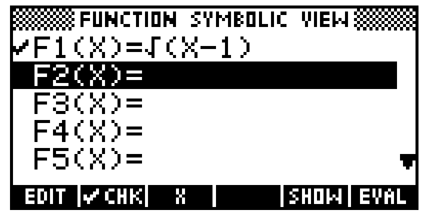
- All the graphs have two points in common. What are they and why are they common?
- Why does an increasing value of *n* mean a lower value of $f(x)=x^n$ between 0 and 1, while it means a higher value for $x>1$?
- What happens for $x<0$? (The scale will need to be changed in the **PLOT SETUP** view.)
- What do the graphs of $y=x^n$ look like for negative integer values of n?

Domains and Composite Functions

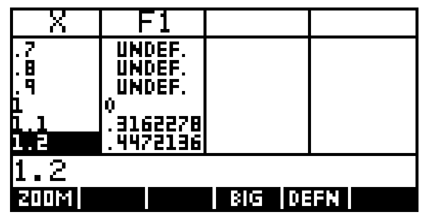
There are a number of ways that the calculator can help with this. Examples are given below but others will no doubt occur to experienced teachers.



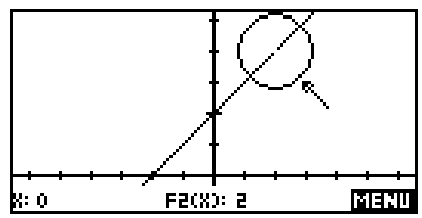
i. Rational functions can be investigated using the **NUM** view. For example, enter the functions **F1(X)=X+2** and **F2(X)=(X2-4)/(X-2)**. Discussion will elicit the fact that they are 'identical' algebraically but what about the point **X=2** in the **NUM** view shown right? This can be used in discussion to introduce the convention of graphing with a 'hole'.



The calculator will display the **Undef.** result whenever the x value in the **NUM** view is outside the domain of the function.



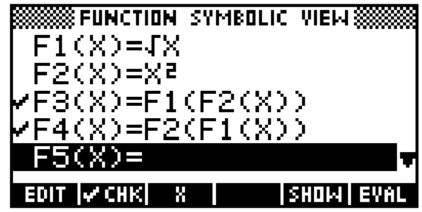
A teacher can also have fun with the class by telling them to un✓CHK **F1(X)** and then to zoom in repeatedly on **X=2** in the **PLOT** view in an effort to "find the hole". They won't of course, but you can enjoy watching them and then discuss why they didn't – perhaps a good way to introduce the idea of limits?



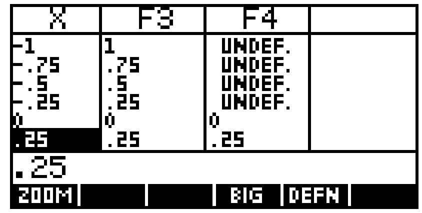
However - there is a trick to this! If you use the default axes of -6.5 to 6.5 then there *will* be a hole (see right) because **X=2** falls on a pixel point and so, because it is undefined, the calculator leaves it out. The missing value means that there is no connection to the pixels on either side.

For this to activity work you need to sabotage their efforts in advance via a scale which does not have **X=2** on a pixel. Starting with a scale like this ensures that subsequent box zooms won't produce the "hole". A good scale is -1 to 6 on both axes and you can rationalize the choice by telling them that it "focuses well on the point we're interested in". They may still sabotage this by choosing their own axes when zooming.

ii. When discussing the concept of a domain, the **NUM** view can be very useful in developing this (see right).

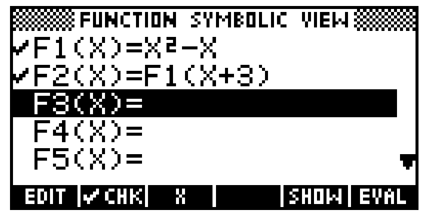


In the **SYMB** view, enter the functions shown right, un✔CHKing the first two non-composite functions. In the **NUM** view shown, I have used the **NUM SETUP** view to set the scale to start at -1 and increase in steps of 0.25.

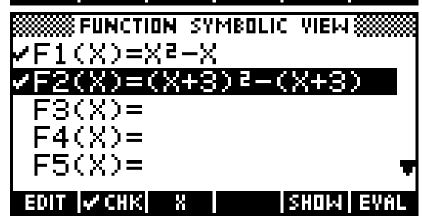


Obviously discussion will now center on why $f(x)=\sqrt{x^2}$ is not the same as $f(x)=x$, and why $f_1\left(f_2(x)\right)$ is not the same as $f_2\left(f_1(x)\right)$ for x<0.

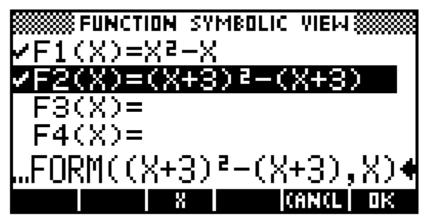
iii. Composite functions can easily be defined, as can be seen in the examples to the right.



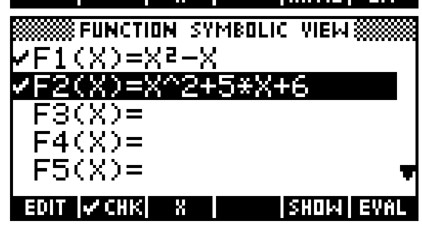
In the first screen shot, **F1(X)=X²-X** and **F2(X)=F1(X+3)** have been entered into the **SYMB** view.



The second, substituted view is obtained by moving the highlight to **F2(X)** and pressing the EVAL button.

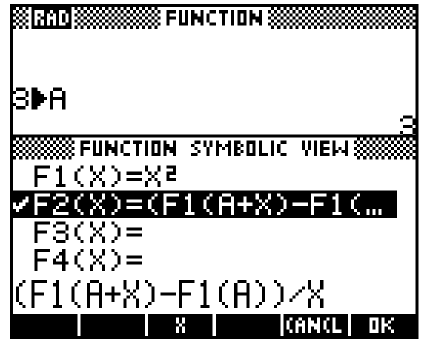


If desirable, you can further simplify using **POLYFORM**. With the highlight on **F2(X)**, press EDIT. Move the highlight to the start of the expression and use the **MATH** button to enter "**POLYFORM(**". Now move to the end and add "**,X)**" to the expression and press OK.



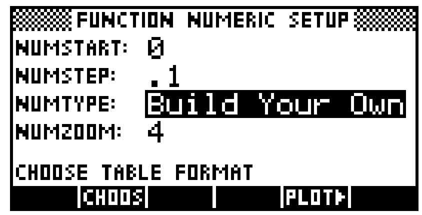
Pressing EVAL again now will give the result shown right.

Gradient at a Point

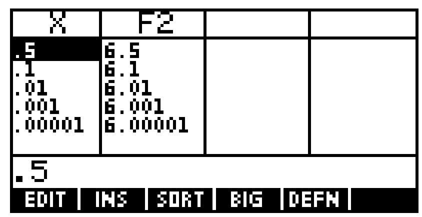


This is best introduced using an aplet called "Chords" downloaded from *The HP HOME View* web site (at http://www.hphomeview.com), but you can also use the Function aplet. If you use the aplet you will find that there is a worksheet supplied with it.

To do it in the Function aplet, enter the function being studied into **F1(X)**. To examine the gradient at x=3, store 3 into A in the **HOME** view as shown right, then return to the **SYMB** view and enter the expression shown right into **F2(X)**.



Change to the **NUM SETUP** view and change the **NumType** to "Build Your Own".

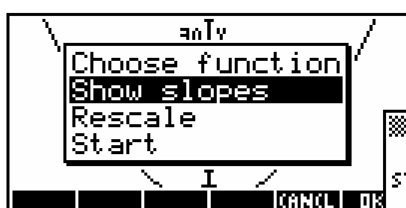


You can now enter successively smaller values for X in the **NUM** view, since X is taking the role of *h* in the expression

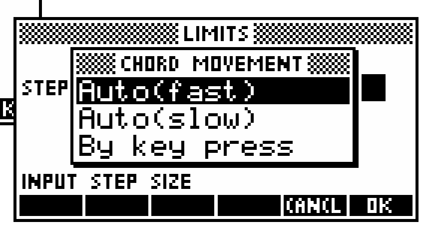
$$f'(a)=\lim_{h\to 0}\frac{f(a+h)-f(a)}{h}.$$

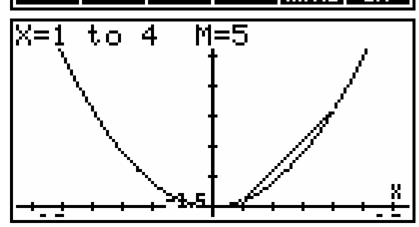
To investigate the gradient at a different point, change back to the **HOME** view, enter a new value into A and then return to the **NUM** view.

The disadvantage of the previous method is that it is not very visual. As mentioned before, an alternative is to use the "Chords" aplet.



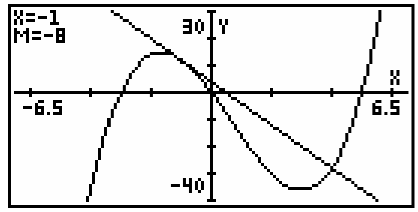
In this aplet, a menu is provided via the **VIEWS** menu to allow students to choose from a list of predefined functions or enter their own.





Once the function has been graphed, the 'Show slopes' option will display an animated series of chords of diminishing length, with the gradient displayed at the top of the screen. As the chord shortens the student can see how this affects the approximation to the gradient at the point chosen.

Gradient Function



Once the concept of gradient at a point has been established the next step is to develop the idea of a gradient function. This can be done via the Function aplet by using the **FCN** *Slope* function which gives the gradient of the graph at the position of the cursor (see page 58).

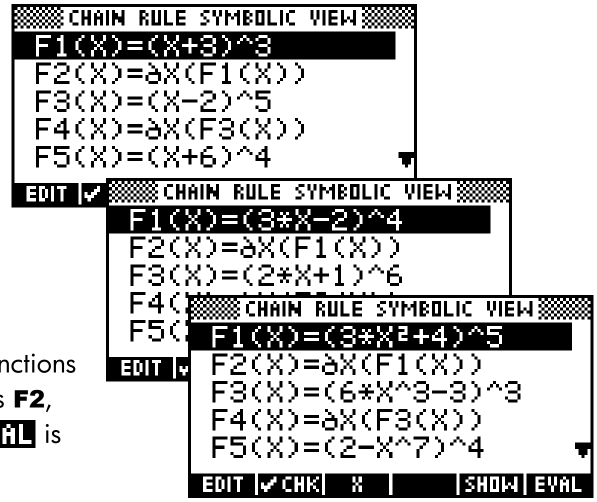
If the teacher has the student enter a function in the SYMB view they can then have the student explore the value of the slope at various values using the **GOTO** function to position the cursor precisely.

The Statistics aplet can be used to help with the process of finding gradient functions once tabular data has been collected giving *x* and *slope(x)* values. If the student enters the data into **C1** and **C2**, they can then set to **2VAR**, plot the data and make a guess as to the appropriate type of function in the **SYMB SETUP** view and then use the curve fitting facilities to find an equation.

Curves of the form $y = mx + b$, $y = ax^2 + bx + c$, $y = ax^3 + bx^2 + cx + d$ and $y = ax^b$ can be fitted using the Statistics aplet and this should be enough for the students to deduce the rule $\frac{d\left(x^n\right)}{dx} = n\,x^{n-1}$ for themselves. It is advisable to ensure that the students are familiar with the process of using the Statistics aplet to find equations before commencing, otherwise the two concepts may interfere and confuse.

The process can be done in a far more visual style using an aplet downloaded from *The HP HOME View* web site (http://www.hphomeview.com) called "Tangent Lines". This aplet will add a moveable tangent line to an existing graph, as shown in the screen shot above, allowing the user to move it along the curve with the gradient displayed at the top left of the screen. There are two worksheets included in the documentation which is bundled with this aplet which will take the student through the process of developing a gradient function.

The Chain Rule



If desirable, an aplet is available from *The HP HOME View* web site (at http://www.hphomeview.com), called "Chain Rule", which will encourage the student to deduce the Chain Rule for themselves.

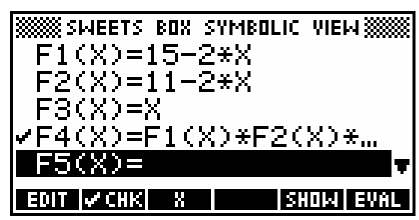
It is pre-loaded with five sets of functions, of increasing complexity, the first three of which are shown right. The functions are loaded into **F1**, **F3**, **F5**, **F7** and **F9**, while the functions **F2**, **F4**, **F6**, **F8** and **F0** contain an expression which, when **EVAL** is pressed, will differentiate the function above.

Through the worksheet which is bundled with the aplet, the student is directed to record the functions and their derivatives and to look for patterns which will allow them to deduce a rule.

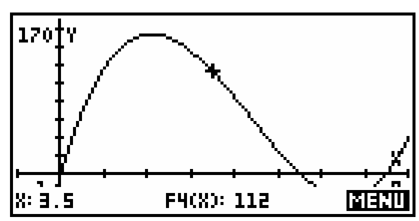
Obviously the student could simply type these functions in themselves but it would be very tedious and the aplet automates the process so as not to obscure the learning.

Optimization

A method which I find to be efficient in introducing the idea of optimization is via the maximization of the volume of an open-topped box.

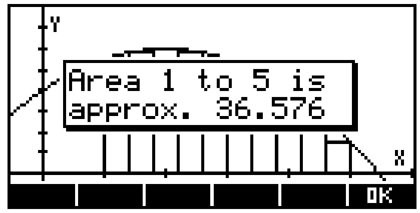


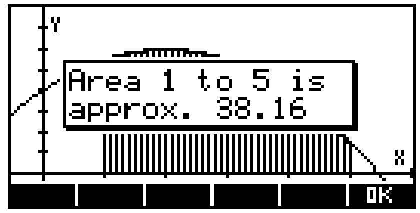
If we start with a sheet of card which is 15cm by 11cm then we can form a box by removing squares from the corners and folding up the sides. I find that it is quite helpful for the students to actually make such a box, choosing for themselves what size square to cut out. They can then explore, using the Function aplet, what cut-out size will produce the maximum volume.



As can be seen above & right, the width, length and height can be entered into **F1**, **F2** and **F3** as functions of the cut-out size **X**. The volume can then be entered into **F4** as **F4(X)=F1*F2*F3** and this function can then be plotted and the maximum found either through successive approximations in the **NUM** view or by using the **FCN** tools in the **PLOT** view.

Area Under Curves



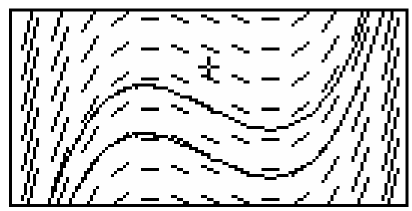


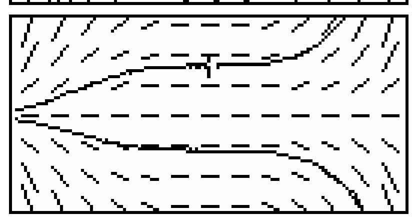
This topic is most easily handled using an aplet from *The HP HOME View* web site (at http://www.hphomeview.com). This aplet, called "Curve Areas" will draw rectangles either over or under a curve or use trapezoids. A number of curves are supplied pre-set but the user can also enter their own. The user can nominate the interval width and the number of rectangles.

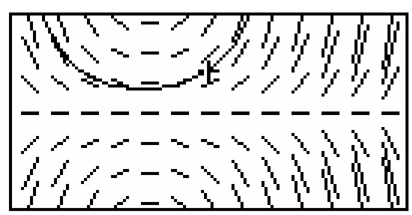
Most importantly, a worksheet is bundled with the aplet which will lead the student through the process of deducing an area function and hence to the anti-differentiation of x^n.

Fields of Slopes and Curve Families

One of the concepts which students find quite difficult to come to grips with is that of sketching a field of slopes from a derivative function and, from this, sketching a family of curves. An aplet from *The HP HOME View* web site (at http://www.hphomeview.com), called "Slope Fields", will assist with this process.





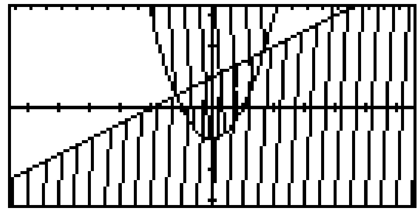


In this aplet the user enters the derivative function into **F1(X)** and then uses the **VIEWS** menu to produce a field of slopes. A cross-hair is projected onto the field which the user can move around. When the user presses **ENTER**, a curve is drawn, starting at that point and projecting to the right and then the left, and following the field of slopes. Repetition of this will illustrate the fact that there are a family of curves, separated by a constant, which all fit the 'description' of the function stored in **F1(X)**.

The screen shots to the right are the result of **F1(X)=X²+1** $\left(\frac{dy}{dx} = x^2 + 1\right)$, **F1(X)=X²*Y** $\left(\frac{dy}{dx} = x^2 y\right)$ and **F1(X)=(X+1)*Y** $\left(\frac{dy}{dx} = (x+1)y\right)$ respectively.

Inequalities

The topic of inequalities is one that is often included in calculus courses, particularly during the study of domains and this is usually extended to graphing intersecting regions such as $\{(x,y): y \le 0.5x+1 \cap y \ge x^2-1\}$.

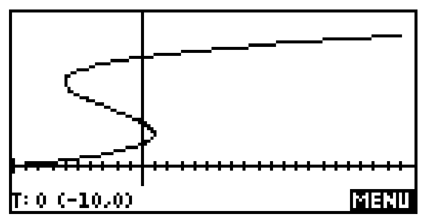


Although the hp 39gs & hp 40gs do not have the in-built ability to plot inequalities, the process is easily handled using an aplet from *The HP HOME View* web site (at http://www.hphomeview.com) called "Inequations".

This aplet allows the user to plot individual or overlapping inequalities but it will not handle functions of the form *y<f(x)* which require a dotted line.

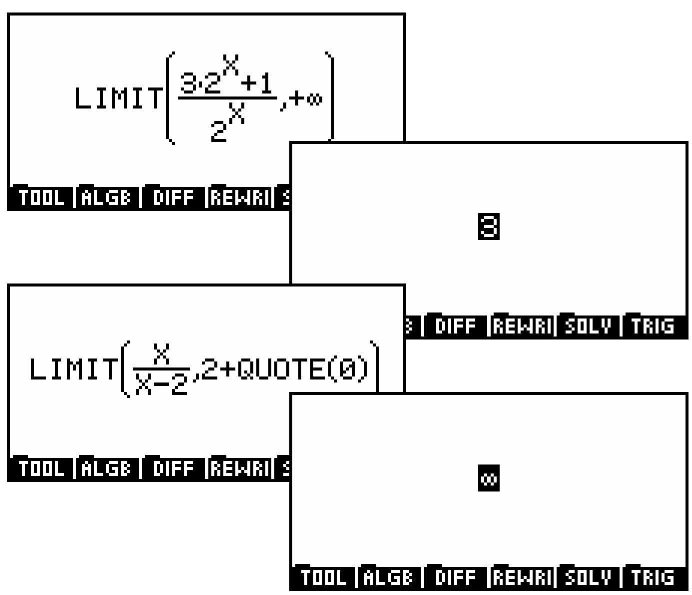
Rectilinear Motion

A topic which is commonly taught as part of any calculus course is rectilinear motion. This can be enhanced by using the Parametric aplet to graphically illustrate the motion of a particle. If this is set up properly then it can be a very helpful teaching aid, as the graph will slow down and speed up as it appears, illustrating the velocity and acceleration of the particle.



See page 96 for a fully worked example of how the Parametric aplet can be used to produce motion graphs of the form shown right. The graph needs to be seen as it is being drawn to appreciate how the particle slows down and speeds up as it passes the turning points. Try it and see.

Limits



For information on exploration of limits in the NUM view of the hp 39gs & hp 40gs and, more importantly, the pitfalls that lie in wait for the unwary, please read the information in the chapter on page 80.

As can be seen right, f or those with an hp 40gs, limits can be evaluated in the CAS using the LIMIT function. See page 343.

The examples shown right are for $\lim_{x \to +\infty} \frac{3 \times 2^x + 1}{2^x}$ and $\lim_{x \to 2^+} \frac{x}{x-2}$.

Piecewise Defined Functions

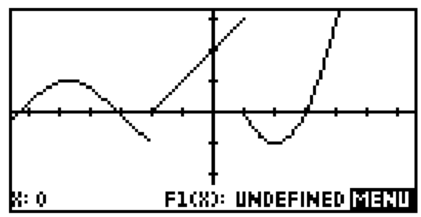
Piecewise defined functions can easily be graphed on the calculator by breaking them up into their components.

For example:

$$f(x)=\begin{cases}\sin(x) & ; \quad x<-2\\ x+2 & ; \quad -2\le x\le 1\\ (x-2)^2-1 & ; \quad x>1\end{cases}$$

Using the Function aplet, we enter three separate component functions. You can obtain the inequality signs from the **CHARS** menu.

F1(X)=SIN(X)/(X<-2)
F2(X)=(X+2)/(-2≤X AND X≤1)
F3(X)=((X-2)2-1)/(X>1)



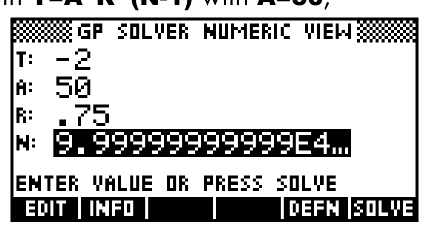
The calculator evaluates the domain as either true (1) or false (0) for each value of x. When it is zero (outside the domain) then dividing by this value causes the function to become undefined and consequently not be graphed. Inside the domain it has no effect.

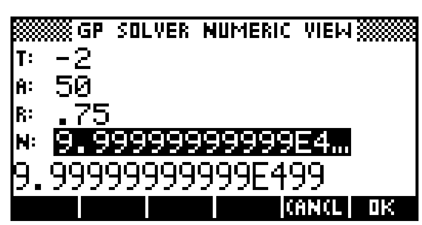
Note: The **AND** is available on the keyboard above the (-) button.

Sequences and Series

Through the Sequence aplet the hp 39gs & hp 40gs provide very flexible tools for the investigation of sequences. These can easily be adapted to investigate series as well. Information and worked examples of using the calculator for evaluation of sequences and series can be found in the chapter "The Expert – Sequences & Series" on page 102.

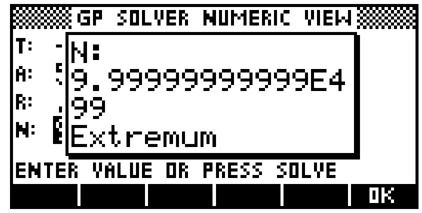
One of the common errors that students make when using the Solve aplet with GPs is to try to solve for solutions which do not exist. For example, a student will try to solve for **N** in **T=A*R^(N-1)** with **A=50**, **R=0.75** & **T= -2**. The result will be firstly that the calculator will seem to freeze while it tries to find a solution which can only be approached asymptotically.





Finally the calculator will give the result as shown right. The problem is that students will tend to misinterpret it as being **N=10**, when in fact it is simply that the calculator has gone as far along the positive x axis as possible and stopped at **MAXREAL** of 1×1^{500} (see second screen shot). It is recommended that the teacher should deliberately provoke this error and follow with class discussion.

Students should also be encouraged to press the **INFO** button after finding a solution since a case like this will give 'Extremum' whereas a correct solution will result in either 'Zero' or 'Sign Reversal'. See the manual for more information.

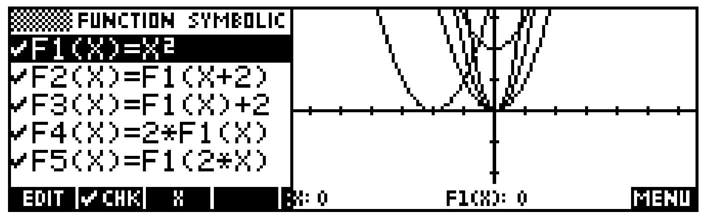


Transformations of Graphs

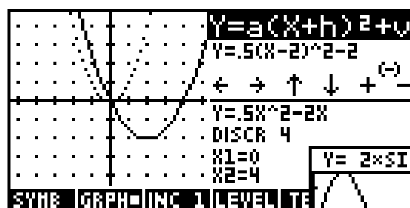
This topic can be handled in a number of ways. One of these is to use the Function aplet without enhancement.

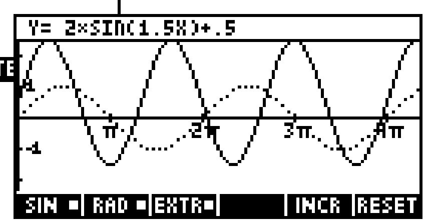
Enter the basic function into **F1(X)**. For example, you might enter **F1(X)=X²**.

You can then enter transformations into the other functions. Some examples, together with the resulting graphs are shown right.



This process will also work with piecewise defined functions which are often the type that are used in examination questions.





There are also two aplets which can aid greatly in this process called "Quad Explorer" and "Trig Explorer" and are built into the calculator. Both of these aplets allow the student to explore the effect of changing parameters on the shape of the graph, one using a quadratic and the other with the sine and cosine curves. Students can choose to manipulate the graph and see the effect on the parameters, or to manipulate the parameters and see the effect on the graph. In Quad Explorer there is even a "test yourself" facility provided which will present the student with examples of quadratics for which they must provide an equation, with visual feedback on incorrect guesses. These are highly recommended. See pages 159 & 162.

44

APPENDIX C: THE CAS ON THE HP 40GS

Introduction

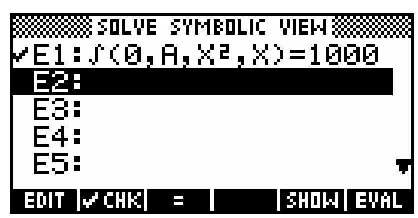
This appendix is intended to give a useful introduction and over view to the user who is new to an hp 40gs. It is *not* intended to fully cover the topic, nor is it intended to serve as a reference text for the advanced user.

For those needing a far more extensive coverage than is available here, I can highly recommend the incredibly detailed text "*Computer Algebra and Mathematics with the hp 40g, Version 1.0*" by Renée de Graeve, Lecturer at the University of Grenoble and founder of the Grenoble IREM. The hp 40g was the immediate predecessor to the hp 40gs and the CAS functions are almost identical. The only significant difference is that on the hp 40gs you exit the CAS by pressing **HOME** rather than using **ON** on the 40g. Additionally, you access the CAS History using the **SYMB** key on the hp 40gs rather than **HOME** on the 40g. This excellent text contains a complete reference of the functions for symbolic calculation, and also demonstrates, using many examples, how to take smart advantage of the calculating power of the hp 40g (or hp 40gs). It can be found at Hewlett Packard's web site (http://www.hp.com/calculators) or on the Help page of *The HP HOME view* (http://www.hphomeview.com).

What is a CAS?

Although you may not have thought about it consciously, you are probably aware that most calculators do not operate with algebra in the same way that a human does. Generally speaking solutions on computers and calculators are found using numeric means. This is often done by using successive approximations, each one coming closer and closer to the final answer.

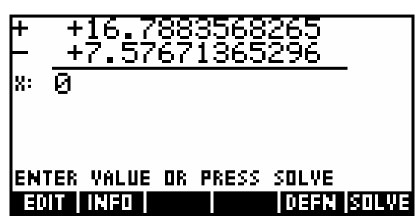
You can see this process at work quite easily using the Solve aplet. All you need to do is enter an equation to solve which is complex enough to require more than a few seconds to solve. If you do this then you can watch the calculator work.



In the Solve aplet, enter the equation

∫(0, A, X², X) = 1000 as shown right.

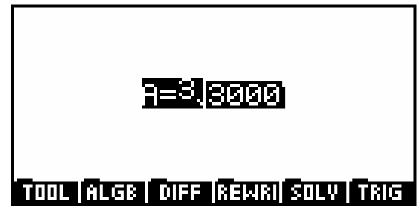
This is equivalent to solving for *a* in the equation: $\int_0^a x^2 \; dx = 1000$.



Change into the **NUM** view, and press **SOLVE** on **A**. As soon as the hour glass symbol appears, press the **SOLVE** button again. You may need to press it more than once before the calculator takes notice, but eventually you should see a view similar to the one right. The two values at the top of the screen represent the calculator's successive approximations to the true solution.

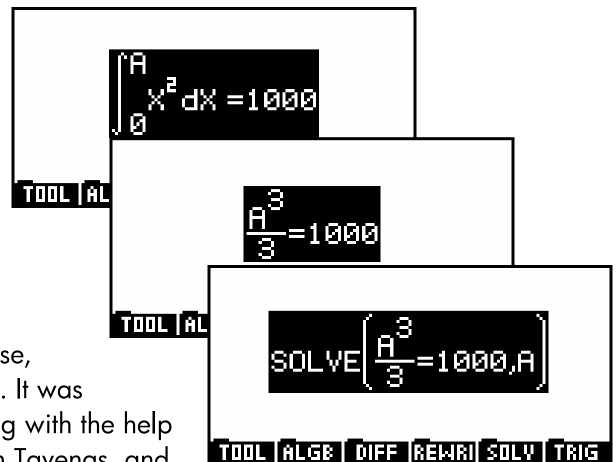
The chances are that one will have a '+' symbol to the left of it, while the other has a '-'. This is telling you that the '+' value is greater than required, while the '-' value is smaller. As you watch you should see the two values converge to the true answer.

But is it? The true answer is actually $\sqrt[3]{3000}$, as is shown right in the hp 40gs CAS. Unless you were alert enough to spot it you probably would not realize that the value supplied was a cube root.



On most calculators there is no way to obtain this exact answer because the calculator doesn't use algebra.

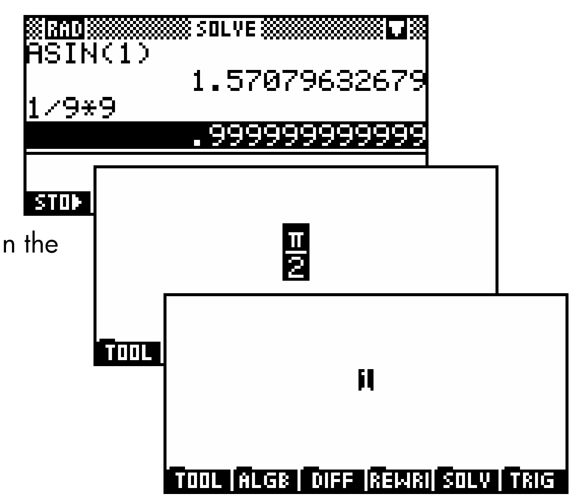
However, the CAS or Computer Algebra System on the hp 40gs *does* use algebra! As you can see in the screen shots to the right, the CAS on the hp 40gs is perfectly capable of giving you the algebraically correct answer, and it does it by following the same rules that you do.



The HP CAS system was created by Bernard Parisse, Université de Grenoble, for the HP 49g calculator. It was improved and adapted for inclusion on the HP 40g with the help of Renée De Graeve, Jean-Yves Avenard and Jean Tavenas, and again adapted for inclusion in the HP49g+, HP48GII and HP 40gs calculators. The HP CAS system offers the user a vast array of functions and abilities as well as an easy user interface which displays equations as they appear on the page.

It will let you perform virtually any mathematical manipulation you might need.

For an example of the fundamental difference between working in the CAS, with its exact mode & infinite precision, and working in the normal **HOME** view, in approximate mode with 10^{-12} precision, you need only consider the two contrasting results of **ASIN(1)** and $\frac{1}{9}\times 9$ shown to the right as they appear in the **HOME** view and in the CAS.



What is the difference between the hp 39g, hp 40g, hp 39g+ and the hp 39gs & hp 40gs

There were two competing sets of requirements at the time that a previous model, the hp 39g, was designed as an upgrade from the original hp 38g. The European market wanted a calculator which had a CAS system but they were highly distrustful of the infra-red communication which was standard on the hp 38g, feeling that it might allow cheating in examinations. The US and Australian markets, on the other hand, wanted the infra-red communications but their educational systems were not yet prepared to allow the power of a CAS into their classrooms, feeling that it did too much of the students' work.

The solution adopted then was to use the same chip for two 'sister' calculators, the hp 39g and the hp 40g. These two models had, respectively, infra-red but no CAS and CAS but no infra-red. They were otherwise identical except for the label below the screen. The calculator's chip was able to detect which model it was in and activate or de-activate the CAS accordingly. Unfortunately a hacker found a way to write a program that would fool the hp 39g into thinking that it was an hp 40g and thus activate the CAS. This caused immense problems in educational markets that required that calculators must not have a CAS and consequently the hp 39g+ was released in 2004. It was an upgraded model of the hp 39g, the differences being vastly greater speed, a new Finance aplet and, most importantly, a chip from which the CAS had been totally removed.

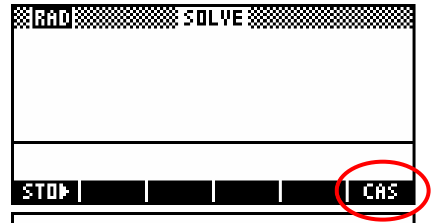
In 2006 the new models, the hp 39gs & hp 40gs were released. These were aimed at the same educational markets as the hp 39g & hp 40g respectively. Again, the hp 39gs has no remnant of the CAS which can be activated. It also had two new aplets, the Triangle Solver and the Linear Solver aplets.

If you own an hp 39gs and you are wondering if you can fool it into believing that it is an hp 40gs and thereby activate the CAS, then please don't try it. If you disable the infra-red by, for example, cutting the internal wires then you won't have an hp 40gs - you will simply have a crippled hp 39gs. Nor is it possible to activate the CAS by means of a special program since it simply does not exist on the chip to be activated.

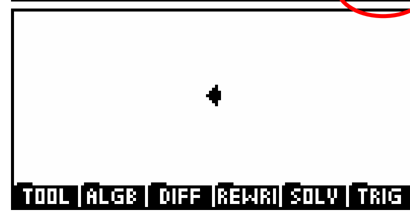
It is also not possible to download the hp 40gs operating system into the flash ROM on the hp 39gs because the extra space required by the CAS means that it physically won't fit into the smaller memory ROM installed on the hp 39gs.

Essentially: Forget it!

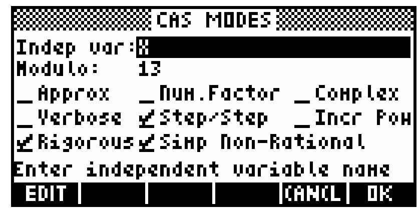
Using the CAS



The first step is to activate the CAS. This is done from the **HOME** view by pressing screen key 6 (**SK6**), labeled **CAS**.



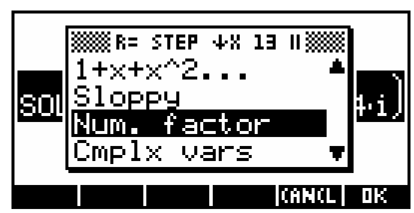
When you do, you will see an empty screen with a cursor in the center and an extensive menu system at the bottom of the screen.



In the examples which follow it will be assumed that the CAS is in its default settings. The **CAS MODES** screen is shown to the right. More detailed information on this is given on page 362.

This screen can be accessed within the CAS in a number of ways:

- from the **HOME** screen, pressing **SHIFT** **CAS**
- by pressing **SHIFT MODES** within the CAS
- by pressing **SHIFT SYMB** within the CAS
- via the **CFG** menu entry shown right. This appears as the first entry on most of the menus within the CAS. The configuration menu that this accesses is not the same screen as that shown above.



To ensure that the CAS is in default mode, enter the CAS and press **SHIFT MODES**. Now press **SHIFT CLEAR**. This will restore default settings.

The current variable

By default, the current variable is **X**, and this is the variable that is entered when you press the X,T,θ button on the keyboard. There are certain functions, such as **SOVLEVX**, which assume that you are solving for or working on the current variable.

It is possible to change the current variable, although this is not advisable unless you know what you are doing. To do this, press the **VARS** button to access the screen shown right, highlight the **namVX** row and press **EDIT**.

For information on this and the other contents of the **VARS** screen, read the CAS manual that came with your calculator. It is not advisable to change entries in this view without being clear on what you are doing since it can alter the behavior of the CAS. Some functions don't work if **X** is not the current variable.

Defining new variables

In addition to the pre-defined variables you can also define your own using the **STORE** command. These new variables can have names that are more than one character long and can contain not only numbers but objects such as algebraic expressions.

For example, **STORE(X²-1,FRED)** would define a new variable called "**FRED**" which will appear in the **VARS** screen. If you now type **FACTOR(FRED)** then the result would be **(X+1)·(X-1)**. Variable names are case sensitive, meaning that "**FRED**" and "**Fred**" would be regarded as two different variables.

To release the memory taken up by a variable you no longer need, use the **UNASSIGN** command. For example, **UNASSIGN(FRED)** will remove the variable and release the memory it takes up. Alternatively, you can enter the **VARS** screen and use the **PURG** (purge) command, which is one of the screen buttons in that view. Just highlight the variable and press **PURG**.

The functions **STORE** and **UNASSIGN** can both be found in the **ALGB** menu.

Entering and editing an expression

When entering an expression the main point to remember is that evaluations and appending of operations are always done to the currently highlighted element. This is most easily seen with an example. A more detailed explanation will then follow of how the CAS editing screen operates when entering and editing expressions.

The task we will be performing in this first example will be to expand and then factorize the expression $(x-4)(x+5)-(x-40)$ over the set of complex numbers. In this first example images of the keys will be used to ensure that you can see exactly what is required. This will not be a general habit for this section.

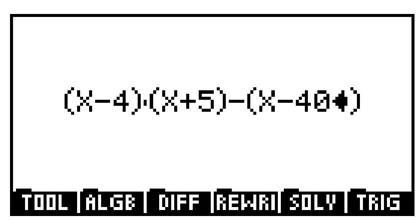
i. Press [X,T,θ], [−], [4] then press **SHIFT** [▲] to highlight this expression.

   Press [×] and the expression will be enclosed automatically in brackets. If you had not highlighted the entire expression first then only the 4 would be multiplied and no brackets would appear. Now add another factor to the expression by pressing

   [(], [X,T,θ], [+], [5].

   Highlight the entire binomial expression by pressing **SHIFT** [▲] then append another expression to this by pressing

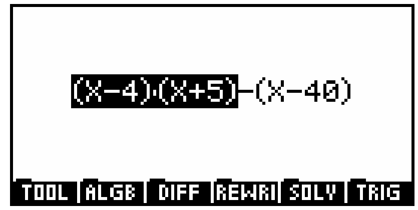
   [−], [(], [X,T,θ], [−], [4], [0].



   At this point the screen should appear as shown right.

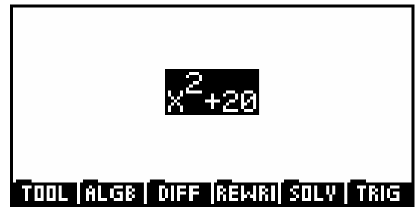
ii. Assume that we want to show working by evaluating the binomial expression separately.

Press [▲], [▲], [▲], [▲], [▲] to highlight the right hand bracket and the subtract, then press [◀] to transfer the highlight to the left hand expression. The screen should appear as shown right.



Press [ENTER] to evaluate this portion of the expression by expanding the brackets without affecting the rest of the expression.

iii. Now simplify the entire expression.



Press [▲], [ENTER]. The result is shown right.

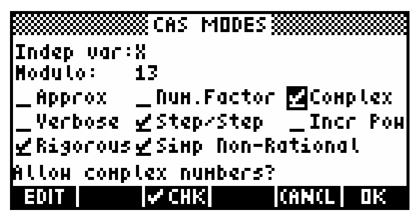
iv. We now wish to factorize this expression. It is already highlighted so choosing the **FACTOR** command will apply it to the entire expression as we require.

Press the [ALGB] screen key, choose **FACTOR** from the menu that appears and press [ENTER]. The screen should now appear as shown to the right.

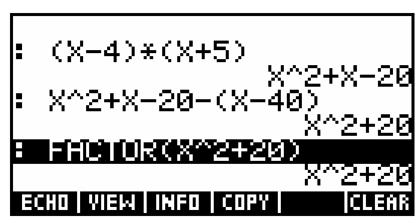
v. Press [ENTER] again to evaluate the highlighted expression.

If your CAS is in its default configuration then the result will simply be a return of the expression x^2+20. The reason for this is that the default setting is to only factorize over the set of real numbers. This needs to be altered using the CAS configuration menu.

vi. To change the CAS configuration, press **SHIFT MODES** and, in the resulting screen, move the highlight to the 'Complex' entry and press [✓CHK] to place a check/tick mark as shown right. Then press [OK] to exit the configuration screen.



vii. To re-use the previous command press [HOME] and highlight the **FACTOR(X^2+20)** line as shown right.



Press [ECHO] to copy it to the screen, replacing the highlighted material currently on the screen.

Alternatively, since this was the last operation applied you could simply press **SHIFT MEMORY** to undo the command and restore the previous screen.

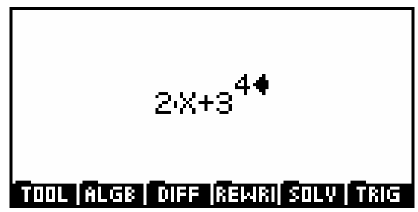
viii. Pressing ENTER this time will result in the screen shown below right which displays the two complex roots.

ix. Press SHIFT, ALPHA, **CLEAR** to clear the highlighted expression, which in this case is the whole CAS editing screen.

In the material on the following page we will explore this editing screen in more detail, looking particularly at why it is necessary to highlight expressions in some cases and not others, and how to move the highlight from one component of the expression to another and how to use special keys to manipulate the expression.

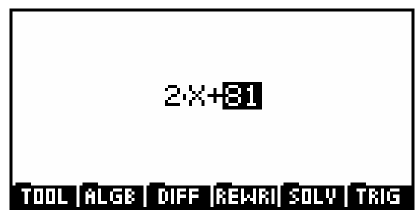
In the **HOME** view press **CAS** and then follow the sequence of keys below:

Press 2 + 3 x^y 4



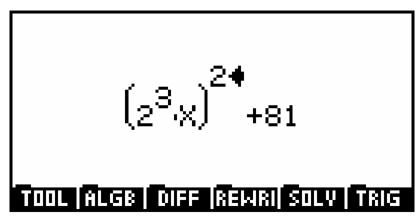
In this case you will notice that the power of 4 was placed so that it applied only to the last character typed; the 3.

Press up arrow twice. Then press **ENTER**.



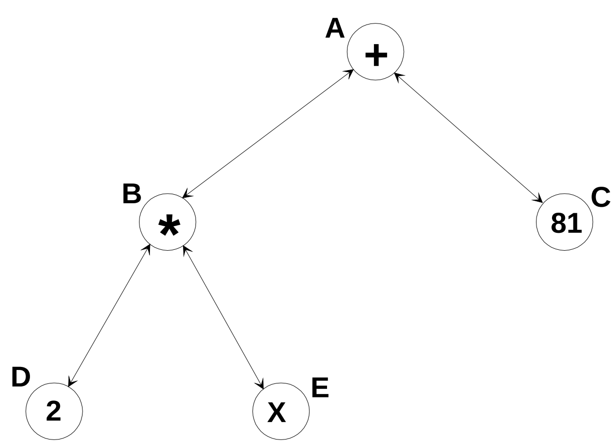
Notice that pressing up arrow caused more and more of the expression to be highlighted. Pressing **ENTER** evaluated the currently highlighted expression of **3^4** to be **81**.

Press left arrow once then down arrow once. Press x^y **3**, then press up arrow three times & finally press x^y **2**.
(*DO NOT use the* **X²** *button for this.)*



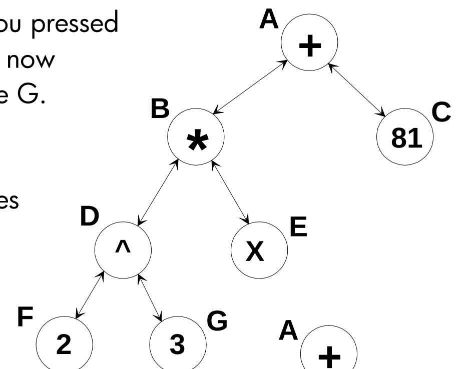
Notice that in each case the power was applied to the currently highlighted element of the expression. Brackets are automatically added as required.

Notice also the movement of the highlight. If you regard the original expression of **2X+81** as a tree of operations as shown right then it may make more sense. When you finished step 2 the highlight was on C, the node containing **81**.



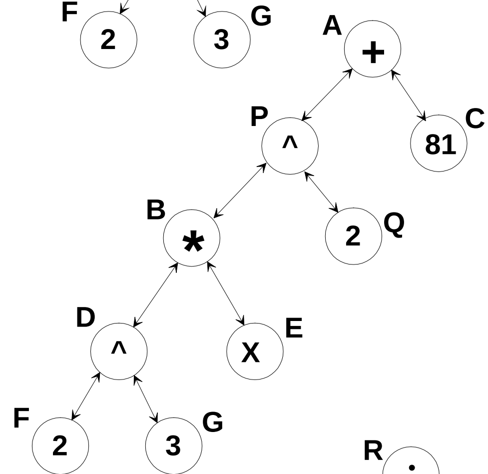
Pressing left arrow moved horizontally from node C to node B. The highlight at that point encompasses everything at or below node B. In this case this is the expression '**2*X**'.

Pressing down arrow at that point moved down the tree. The default is to move to the left-most node D. This meant that the **2** was highlighted and so when you pressed [x^y] **3** it was that expression which was cubed. At that point the tree now appears as shown right with the CAS in 'typing/editing mode' on node G.



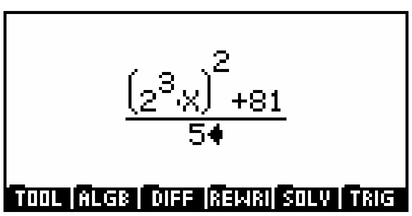
Pressing up arrow three times placed the highlight successively on nodes G, then D, then B. When [x^y] **2** was pressed it squared the entire expression defined by node B which was '$\mathbf{2^3X}$'. This necessitated brackets and so they were automatically added.

The result was the tree shown to the right, with nodes P and Q added below node A. You may notice that it is heavily canted to the left and this tends to be fairly typical of the way we generally write expressions.

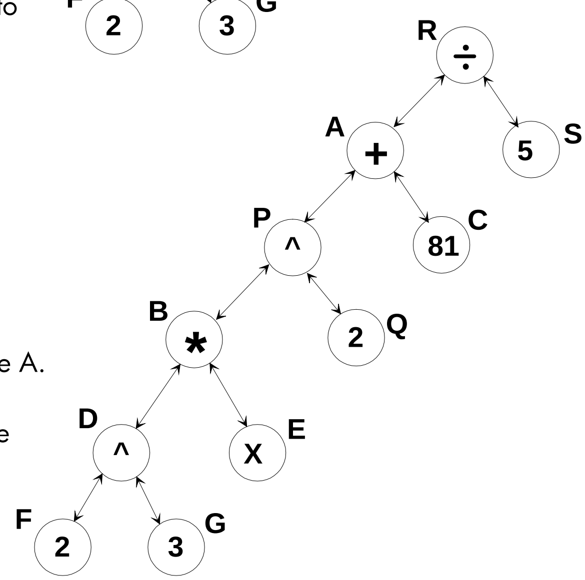


Now press up arrow three times and then divide by **5**.

The three up arrows moved the highlight up to node A, highlighting the entire expression. Dividing by 5 therefore applies to the entire expression, with the result shown below.

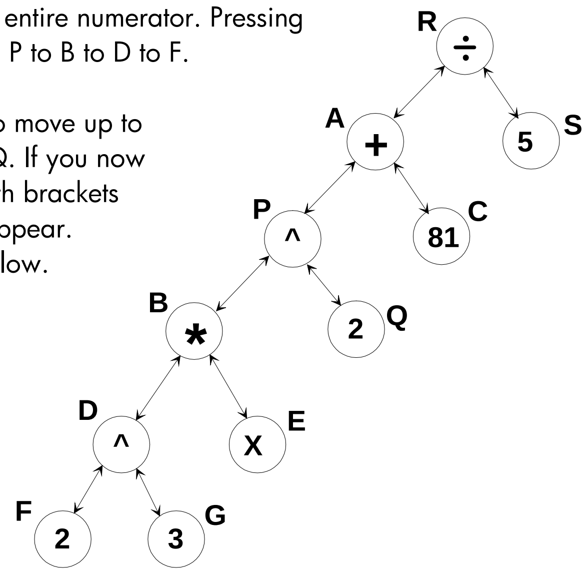


The new tree is shown below with nodes R and S added above A.



Although it is not strictly necessary for you to understand or use this concept of a tree of operations you may find that it will help you to follow why the highlight behaves as it does as it moves around. A final example may help with the visualization.

After typing the 5, press up arrow once to highlight that node S. If you now press left arrow you will find that the highlight will jump horizontally to node A, highlighting the entire numerator. Pressing down arrow four times moves down through the tree from A to P to B to D to F.

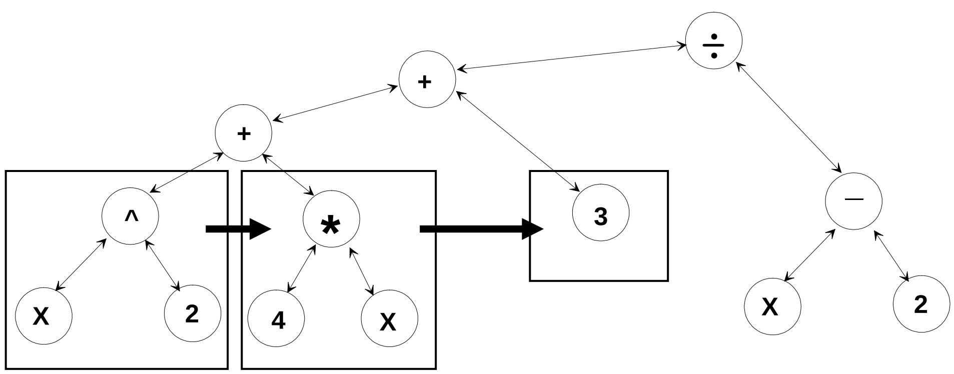


To access and change the power of 2, press up arrow twice to move up to node B, then press right arrow to move from node B to node Q. If you now press **+** and **3** you will find that this is added to the power, with brackets applied as required. Try redrawing the tree as it would now appear. Node Q will then become an addition with two new nodes below.

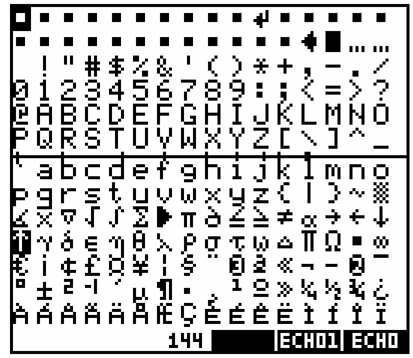
Finally, exit the CAS by pressing **HOME.**

Note: There can be a problem with the way that the **X²** button is handled. If you try going back through the same exercise but pressing the **X²** button at step 3 you will find that this makes it impossible to access and edit the power because the **X²** operation is stored differently. This is purely a personal preference generally.

Another example of a tree for $\dfrac{x^2+4x+3}{x-2}$ is given below to illustrate the result of having three terms in part of the expression. It can be seen that despite having three terms the result is still a binary tree. The squares show the terms which will be highlighted in turn if the x^2 term is highlighted and then the right arrow is pressed.



Special characters



As in the **HOME** view, special characters such as inequalities are available from the **CHARS** view, although the appearance of the **CHARS** view is somewhat different as can be seen right. There are no page up/down buttons, which makes it more difficult to move through. The initial two rows are invalid characters that can't be used – exactly why they were included is not clear.

Pressing **ECHO1** will echo the single character under the curser, closing the **CHARS** view and returning immediately to the CAS editing view. Pressing **ECHO** stores the character to a buffer and allows you to continue selecting more. You can then either press **ON** to exit or use **ECHO1** for the final character. If the character you echo is inappropriate for the situation in the CAS view then it may be rejected.

Using the **CHARS** view is not the most convenient method and some commonly used characters are provided via special keyboard shortcuts.

These are:
- **SHIFT 0** – inserts ∞
- **SHIFT 1** – inserts i ($\sqrt{-1}$)
- **SHIFT 3** – inserts π
- **SHIFT 5** – inserts $<$
- **SHIFT 6** – inserts $>$
- **SHIFT 8** – inserts $\leq$
- **SHIFT 9** – inserts $\geq$

The **(-)** button also has a special function in the CAS. If you have entered a variable or constant, pressing **(-)** will insert a negative character before it even if the cursor is currently to the right of the variable or constant. Pressing **(-)** again will change this to a + (positive) sign.

For example, pressing **SHIFT 3** then **(-)** **(-)** will result in $+\infty$. This can be very useful when working with limits or integrals.

Special editing commands – Undo, multi-select & swap

Unlike most calculators the CAS editing screen has an undo function. If you have performed some operation that was incorrect then pressing **SHIFT MEMORY** will undo the operation. Unlike programs on the PC which have more memory to work with and so allow multiple levels of undo, this will only undo a single operation. However, this can be very convenient at times.

We have seen how the arrow keys can be used to move the highlight within the expression being edited and to select various parts of it. If you use the SHIFT key then this modifies the results in various ways.

i. **SHIFT** ▼

Pressing **SHIFT** ▼ will move the selection immediately to the bottom left corner of the branch of the editing tree currently being worked on. This is best seen by experimentation – effectively, if a number of terms are currently selected, the selection jumps to the single term at the bottom of that branch.

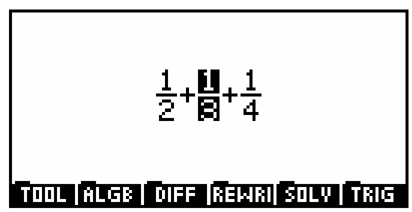
ii. **SHIFT** ▲

Pressing **SHIFT** ▲ will move the selection to the top of the tree, effectively selecting the entire contents of the CAS editing screen.

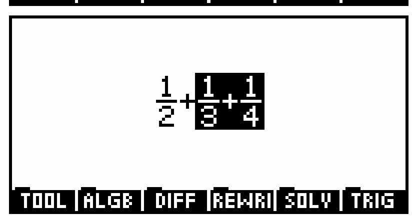
iii. **SHIFT** ▶

Pressing **SHIFT** ▶ behaves in the same way as it does in a word processor on a PC. The current selection is extended to the right, encompassing one more term for each press.

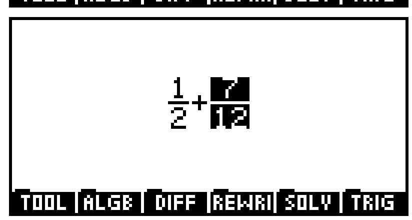
For example, if the current screen showed:



… then pressing **SHIFT** ▶ will result in:

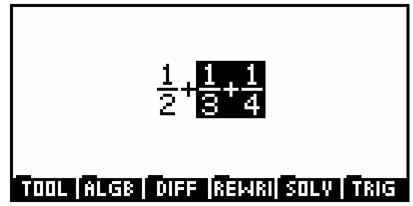


This allows you evaluate or to apply functions to parts of an expression without having to do it to the entire thing. For example, pressing **ENTER** at this point will result in:

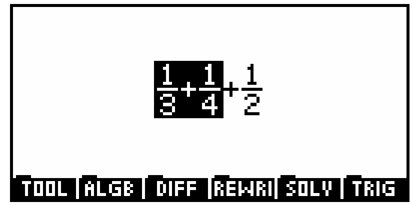


iv. **SHIFT** ◄

Pressing **SHIFT** ◄ swaps the currently highlighted/selected branch for the one on the immediate left.



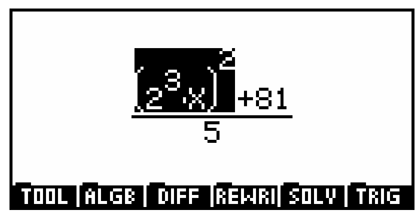
For example, suppose you have selected the middle term and then extended the selection using **SHIFT** ► as shown right:



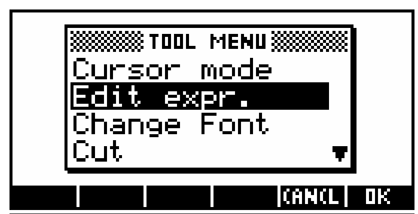
… then pressing **SHIFT** ► will result in:

The main use for this is when you want to substitute a previous result into a new expression but the terms are not in the right order to do so. See page 352 for an example of this (specifically part 5 of the example).

In-line editing mode

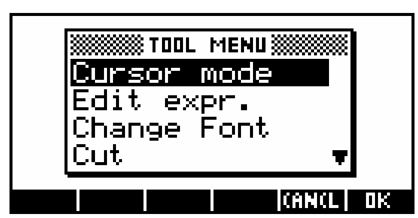


If you find that you are not able to edit part of an expression in exactly the way you need, or if you have entered an operation at the wrong level and are having trouble removing it, then you can highlight and edit an expression 'in-line' as if you were entering it in the calculator's normal **HOME** view.



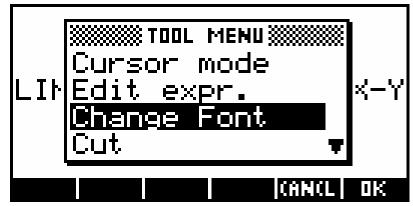
For example, highlight part of an expression as shown right. Now press SK1 (**TOOL**) and choose 'Edit expr.' from the menu. You can now perform any editing you require and the result will be inserted into the full expression at that point.

Cursor mode



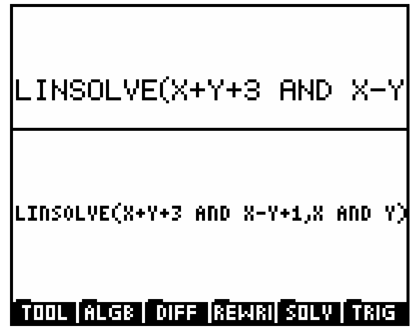
Cursor mode can be used to select portions of an expression by drawing a box around it. From the **TOOL** menu, select 'Cursor mode'. Use the arrow keys to move a cursor around on the screen. As you do so it places a box around different parts of the expression that are selectable. Press **ENTER** to select that section. Unfortunately, it is sometimes not as easy as it looks to select just that small portion of the entire expression.

Changing Font

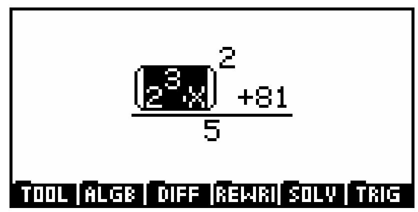


Although the default font is very easy to read, it is quite large and often makes parts of the expression or result extend off the screen. Changing to a small font can help with this, at the cost of making the characters a little more difficult to read. The **Change Font** command is found on the **TOOL** menu. It is a toggle command – it changes to small font if in large mode and vice versa.

Contrasting screens are shown to the right:

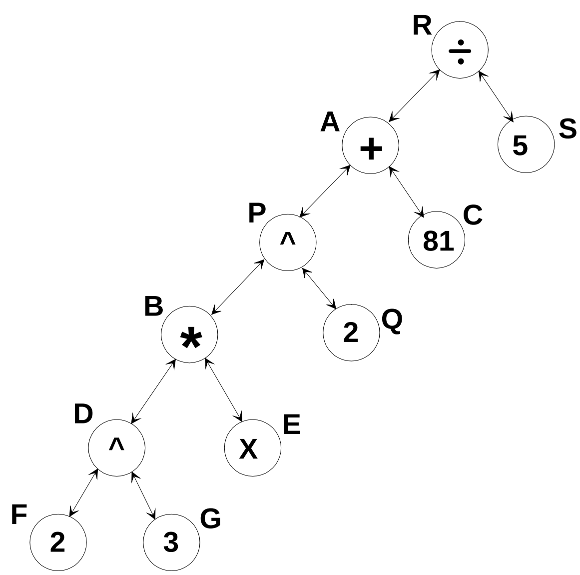


Erasing, copying, cutting and pasting



When you press the **DEL** button in the CAS editor the effect is basically to remove nodes of the tree. The first node deleted is the one furthest right in the currently highlighted section.

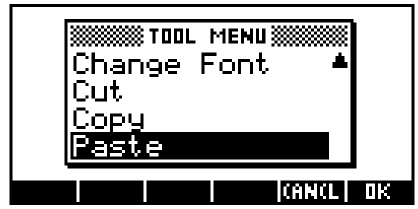
For example, if the highlight was as shown in the screen above, then the current focus would be on node B in the tree shown right. Successively pressing **DEL** would begin at node E, deleting first the node's contents then the operation (multiply) which connected it to the tree. Try it and see.



This process is best seen with experimentation. My experience has been that if you are simply wishing to edit a small portion of the expression then the best method may be to use the 'Edit Expr.' command in the **TOOL** menu. The tree structure used by the editor means that although it is usually very easy to add new operations and expressions within an existing one, it can sometimes be frustratingly difficult.

If you want to delete the entire expression then the simplest method is to press **HOME**, exit the CAS and then re-enter it with a blank screen. Alternatively you can highlight part or all of the expression and then press **SHIFT ALPHA CLEAR**. The highlighted section will be cleared.

Cutting and pasting of all or part of an expression can be easily done using the **TOOL** menu. This provides access to commands of **Cut**, **Copy** and **Paste** which behave in exactly the same manner as they do in any word processor.



Simply highlight the relevant portion of the expression first and then either cut or copy it. Then move to the new position and paste it. As you can imagine from the tree structure the results of a paste can sometimes be unexpected but generally they will be satisfactory.

The CAS HOME History

The CAS has its own history which is essentially similar in its behavior to the normal **HOME** History. If you press **SYMB** while in the CAS then you will see something similar to the view on the right, with all previous calculations and results recorded.

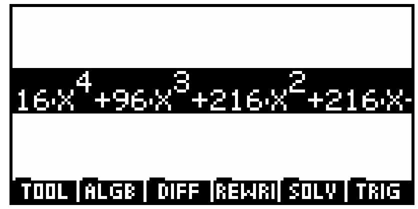
As with the normal History it is worth deleting the contents regularly by pressing **CLEAR** if the memory is not to be gradually used up. Alternatively you can access the Memory Manager (see page 30) and clear the CAS History there.

To re-use an expression just highlight it and press **ECHO**. The expression will replace whatever selection is currently highlighted in the main editing window, or be inserted at the site of the cursor if nothing is highlighted. Alternatively, pressing **COPY** will copy the expression to the clipboard so that you can then paste it using the **TOOL** menu.

Viewing results

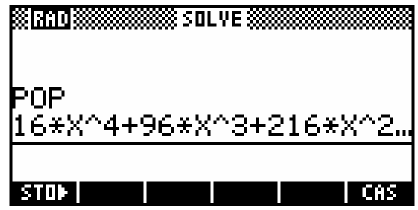
When the CAS presents its results it is quite common for them to extend off the edge of the screen. We have already seen earlier that the font can be changed but, alternatively, pressing the **VIEWS** button will present the expression in a form that can be scrolled left/right and, if needed, up/down.

The PUSH and POP commands

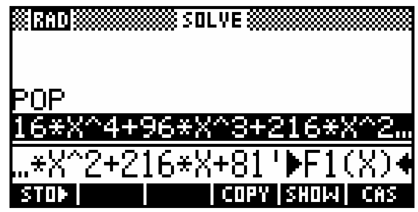


Occasionally it is desirable to transfer results from the normal **HOME** view to the CAS screen or vice versa.

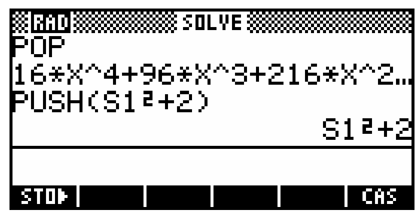
This is done using the **PUSH** and **POP** commands.



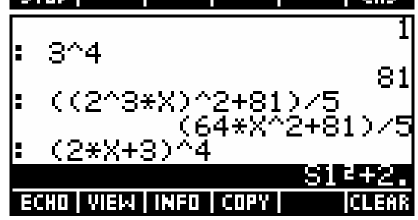
Suppose we have just expanded $(2x+3)^4$ in the CAS, as shown right. If we press **HOME** to exit the CAS and then type **POP** in the **HOME** view then the result will be retrieved to the **HOME** screen as shown. The use of the **POP** command erases the last line of the CAS History so using it a second time will generally produce different results.



In this case we might wish to also paste the result into the Function aplet. Unfortunately the **POP** command only works in the **HOME** view and not in the Function aplet. However you can use **COPY** to retrieve the expression, enclosing it in single quotes from the **CHARS** view and then storing it (▶) into **F1(X)** or whichever is desired as shown right.



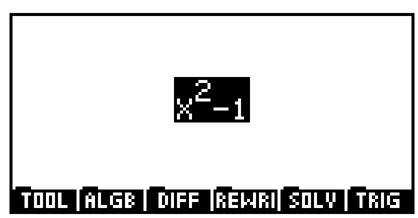
The reverse process is also possible using **PUSH** but it is more limited in that you must use the symbolic variable **S1**. When you then press **CAS** you will find that seemingly there has been no result. However, when you access the CAS History by pressing **SYMB** you will find that the expression has been added to the tail of the History and can now be transferred to the editing screen using **ECHO**.



The drawback is that it comes through still using **S1** as the variable and this may cause problems later. In general the **PUSH** command is not really very useful.

Pasting to an aplet

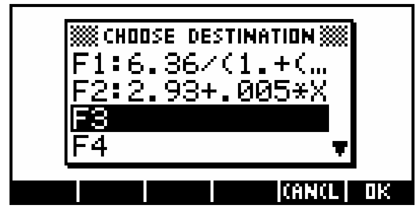
As mentioned above, one method of transferring CAS results to a normal aplet such as Function is to use the **POP** command. However, for graphing results, there is an even easier method - simply press **PLOT**.



Suppose that we have a result in the CAS editor as shown right.

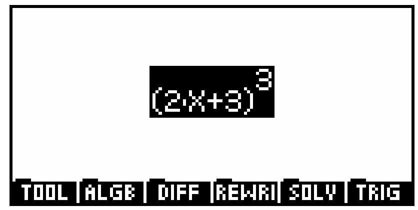
Pressing the **PLOT** button will result in the menu shown in the second screen.

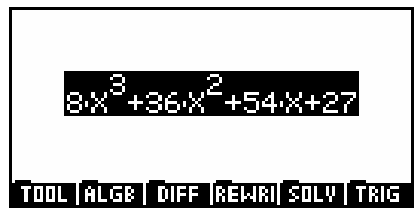
If you choose the Function aplet then you will be asked to nominate a destination. The current contents of each function is shown to allow you to choose whether to overwrite or not. All you need then do is exit the CAS and enter the Function aplet. You will then need to manually **✔CHK** the new function if you want to **PLOT** it. See page 342.



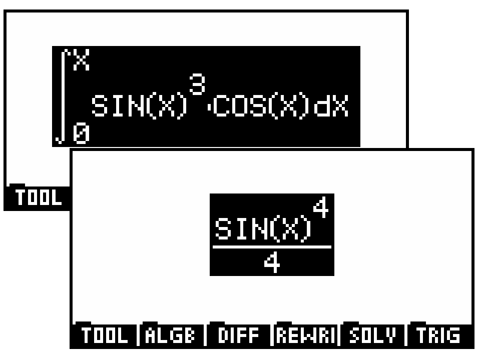
Evaluating algebraic expressions

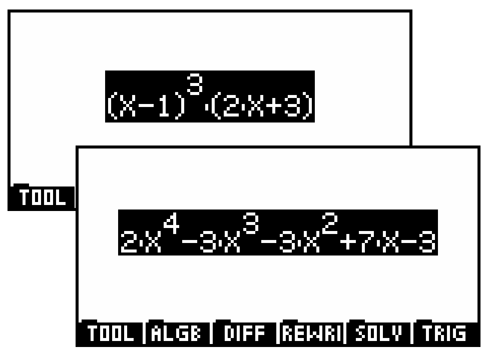
When an expression is highlighted, pressing **ENTER** will cause it to be algebraically evaluated and any functions to be applied accordingly. For example, if you highlight **(2x+3)^3** as shown right then pressing **ENTER** will give the result shown.



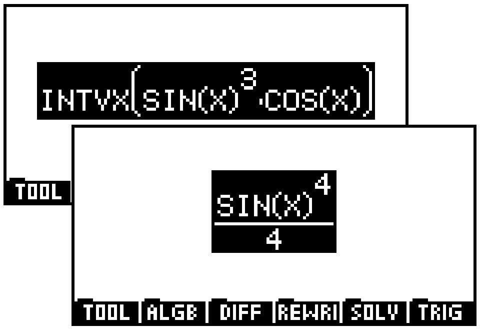


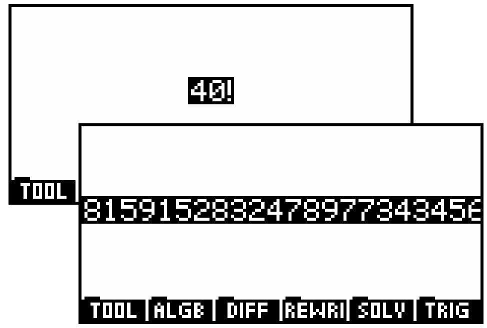
Further examples are shown below. In each case it is only the highlighted expression which is evaluated. Depending on the settings of the **CAS** configuration there may or may not be intermediate steps displayed between those shown.





Notice the lack of a '+c' indefinite constant in the integration result. Here, this is because we are using the definite integral (see page 73 and the page following). A better alternative is to use the **INTVX** function as shown below, even though it still does not add the '+c' (see page 73 for reasons).





Calculator Tip

The result of the **40!** example above extends off the edge of the screen and pressing will not scroll it. If you press the **VIEWS** button, then you will find that it can now be scrolled and the entire value seen - all 48 digits of it! The amazing abilities of the CAS are such that even the entire results of something like **200!** can be seen.

Examples using the CAS

In these examples we will begin with exercises which demonstrate the basic abilities of the CAS to simplify expressions and then move on to the use of the functions available through the various menus. In the initial examples the exact keystrokes will be supplied but in later ones this may not always be the case.

Example 1: Simplifying a fraction with working

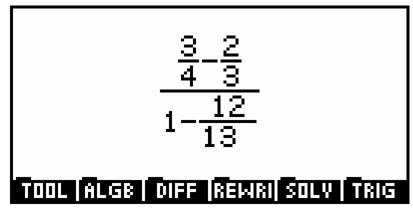
$$\frac{\frac{3}{4}-\frac{2}{3}}{1-\frac{12}{13}}$$

Suppose you are required to simplify the expression shown right, giving your answer as a proper fraction.

i. Begin by entering the top pair of fractions:

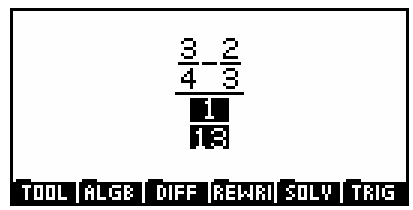
   3 [÷] **4 SHIFT** ▲ [−] **2** [÷] **3**
   SHIFT ▲ [÷] **1** [−] **12** [÷] **13**

   At this point the screen should appear as shown above right.

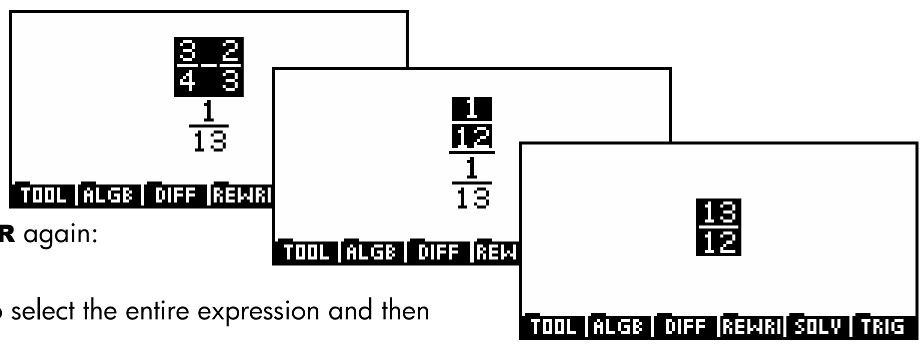


ii. We will now simplify selectively in order to be able to record working, beginning with the denominator:

   ▲ ▲ ▲ ▲ **ENTER**



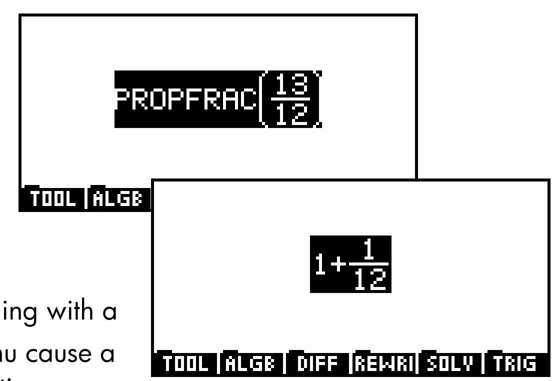
iii. Because the entire denominator is selected, pressing ◄ will now select the numerator. To evaluate this, press **ENTER** again:



iv. Now press **SHIFT** ▲ to select the entire expression and then **ENTER** to evaluate it:

v. To obtain a mixed fraction we use the function **PROPFRAC**, designed to transform improper fractions to mixed. The fraction is already selected, so choosing the fraction now will apply it:

   SHIFT CMDS 7 ► ► **ENTER ENTER**

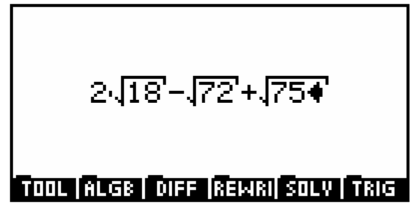


Note: The **7** above causes a jump to the first function beginning with a **P**, the letter on the **7**. The ► arrows within any menu cause a page down. The alternative is just to scroll to the function.

Example 2: Simplifying surds

Simplify the surd expression: $2\sqrt{18} - \sqrt{72} + \sqrt{75}$

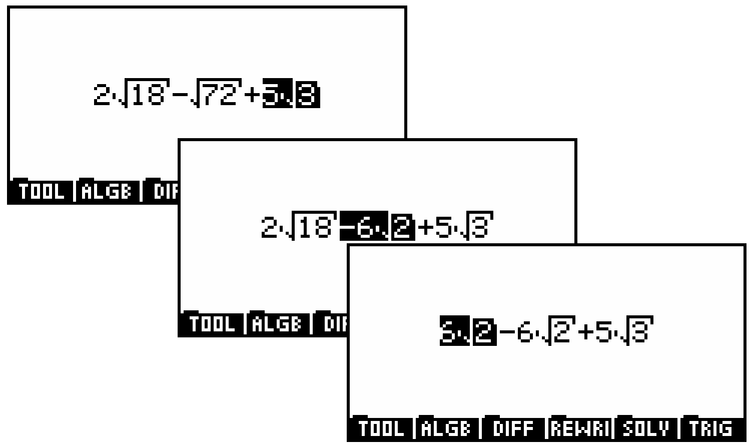
i. Begin by entering the expression:



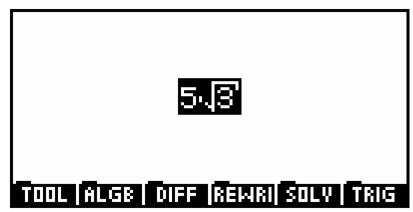
2 SHIFT √ 18 SHIFT ▲ [−] SHIFT √ 72
▲ ▲ ▲ [+] SHIFT √ 75

ii. Now simplify each surd in turn (assuming working is desired):

▲ ▲ ENTER
◄ ENTER ◄ ENTER



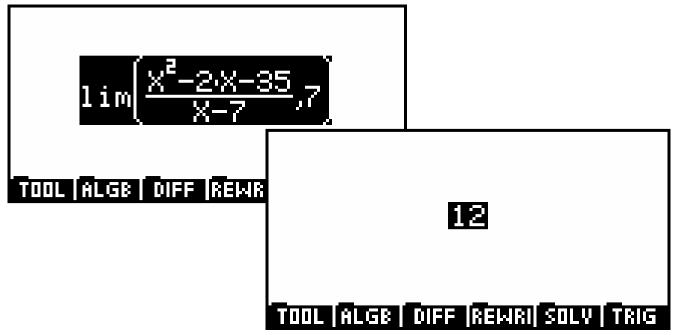
iii. Finally, select the entire expression with SHIFT ▲ and press **ENTER** to simplify:



iv. If you want the result as a decimal, press **NUM**. Pressing **SHIFT NUM** will cause the calculator to analyze the decimal and re-instate the surd.

There are two ways that functions can be used in the CAS. The first is to use them as the expression is entered. In this method the order is to choose the function and then to fill in the parameters required. The second is to apply a function to all or part of an expression that has perhaps resulted from a previous calculation or been typed in first. If you highlight an expression and then choose a function the expression will appear as the first parameter in the function. A list of the functions in their various menus is given on page 358.

Example 3: Using lim



i) Find $\lim_{x \to 7} \frac{x^2 - 2x - 35}{x - 7}$.

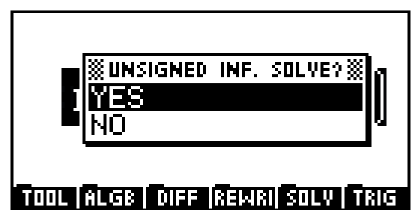
The sequence of keys for this is…

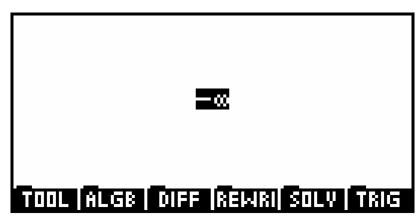
DIFF scroll to **lim** and press **ENTER**
X,T,θ x² **- 2** X,T,θ **- 35** ▲ ▲ ▲
÷ X,T,θ **- 7** ▶ **7** ▲ ▲ **ENTER**

ii) Find $\lim_{x \to 2^-} \frac{1}{x - 2}$.

The sequence of keys for this is…

DIFF scroll to **lim** then **ENTER 1** ÷ X,T,θ **- 2**
▶ **MATH** ▶ scroll to **QUOTE ENTER 2 - 0**
SHIFT ▲ **ENTER ENTER**





Some notes:

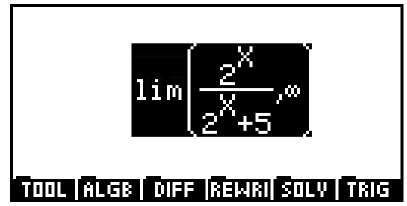
1. The limit approached from above would be entered as **QUOTE(2 + 0)**.

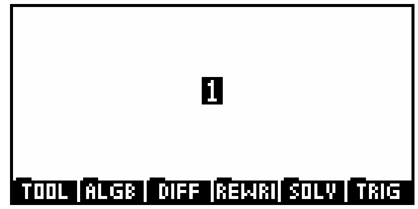
2. The use of the **QUOTE** function forces the CAS to treat the "**2 - 0**" as an algebraic entity rather than immediately evaluating it as simply "2". Thus the result of $-\infty$.

3. The two functions **LIMIT** and **lim** are the same. On the old model hp 40g the only function was **LIMIT** but **lim** was added on the 40gs. Because it was a late entry it appears one way in some menus and as the alternative in the others. Functionally there is no difference so why it was added is not clear.

iii) Find $\lim_{x\to\infty}\frac{2^x}{2^x+5}$

Limits to infinity are also permitted using the **lim** function, with infinity entered using the shortcut **SHIFT 0**.





DIFF scroll to **lim** and press **ENTER**

[2] [x^y] [X,T,θ] [▲] [▲] [÷] [2] [x^y] [X,T,θ]

[▲] [▲] [+] [5] [▲] [▲] [▲] [▶]

[SHIFT] [0] [SHIFT] [▲] **ENTER**

Example 4: Factorizing expressions

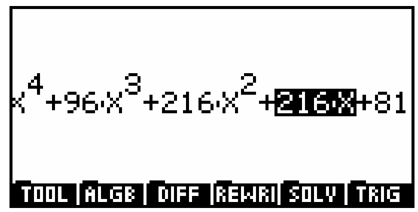
If you highlight an expression such as $(2x+3)^4$ and press **ENTER** then the CAS will expand the bracket. Since the result extends beyond the screen we will scroll through it using the arrow keys. The results can be factorized again using the **COLLECT** function.

In this example we will also illustrate the use of the CAS History to fetch a previous calculation.

The sequence of keys for this is…

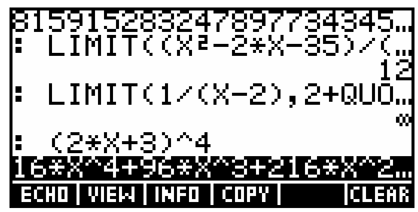
Expansion…

2 X,T,θ **+** **3** X,T,θ **-** **35** ▲ ▲
x^y **4** ▲ ▲ **ENTER**



Examination…

▼ ► ► ► ►

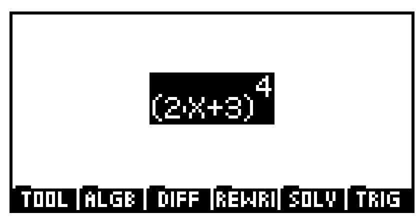


Note: Alternatively you can press **VIEW** and scroll through the result in the view screen. This is equivalent to the **SHOW** command in the **HOME** screen.

Factorization…

Press ▲ until the entire polynomial is highlighted (or press **SHIFT** ▲). While the required expression is highlighted, fetch the **COLLECT** function from the **ALGB** menu.

ALGB **ENTER** **ENTER**



Note: Firstly, remember that the extent of the highlight plays a very important role. Only the highlighted section of the polynomial will be included in the **COLLECT** brackets. This allows partial factorizations if desired. In this case the entire expression should be highlighted before **COLLECT** is inserted.

Secondly, when **ENTER** is pressed the evaluation is done on whatever is highlighted. It is necessary for the entire expression, including **COLLECT**, to be highlighted if it is to be evaluated. If nothing is highlighted then the entire screen is evaluated.

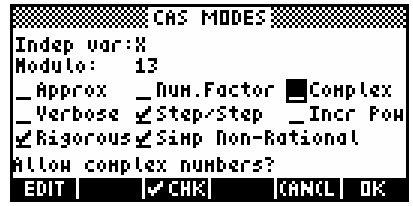
Example 5:   Solving equations

Solve the equation $x^4 - 1 = 3$, giving

i) real solutions and

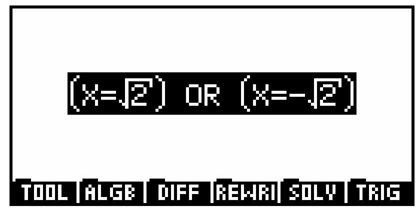
ii) complex solutions.

From within the CAS, press **SHIFT MODES** to access the configuration menu and ensure that you are in real mode by, if necessary, un-✔CHKing **Complex mode**, as shown right.

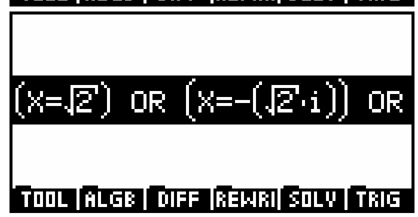


Now use the **SOLVEVX** function, typing:

The **SOLVEVX** function assumes that the active variable is being used. The default active variable is **X** and if no **=** sign is included then the expression is assumed to be equal to zero. The related function of **SOLVE** has a second parameter which allows you to define the variable to be solved for if it is not **X**.



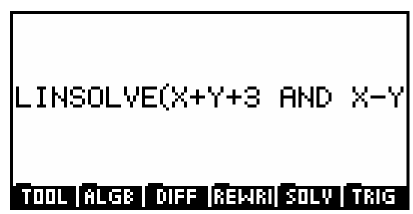
Changing to **Complex mode** in the configuration menu will allow you to find the other two imaginary solutions as shown right.



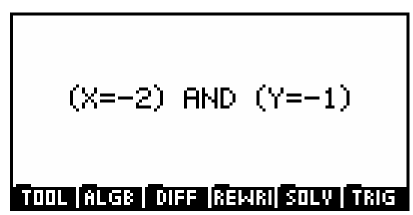
Example 6:   Solving simultaneous equations

Solve the following system of equations: $\begin{cases} x + y = -3 \\ x = y - 1 \end{cases}$

In the CAS, enter: **LINSOLVE(X+Y+3 AND X–Y+1, X AND Y)**.



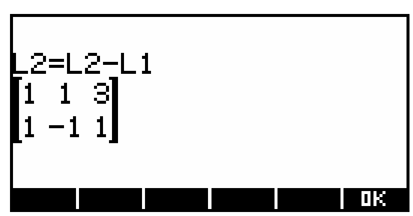
Note that when entering the two equations you must highlight the first one before pressing **AND** (obtained via **SHIFT (-)**). If you don't do this then the CAS will assume that the **AND** applies only to the 3 and place brackets incorrectly accordingly.



Alternatively you can use **X+Y=-3 AND X=Y-1** instead of the expressions. Again, you must highlight the entire first equation before adding the **AND**.

The result is: **(X=–2) AND (Y=–1)**

LINSOLVE can be used for any number of simultaneous variables. If you set **Step by step** mode in the configuration menu then the CAS will show the solution process using row manipulation of an augmented matrix.



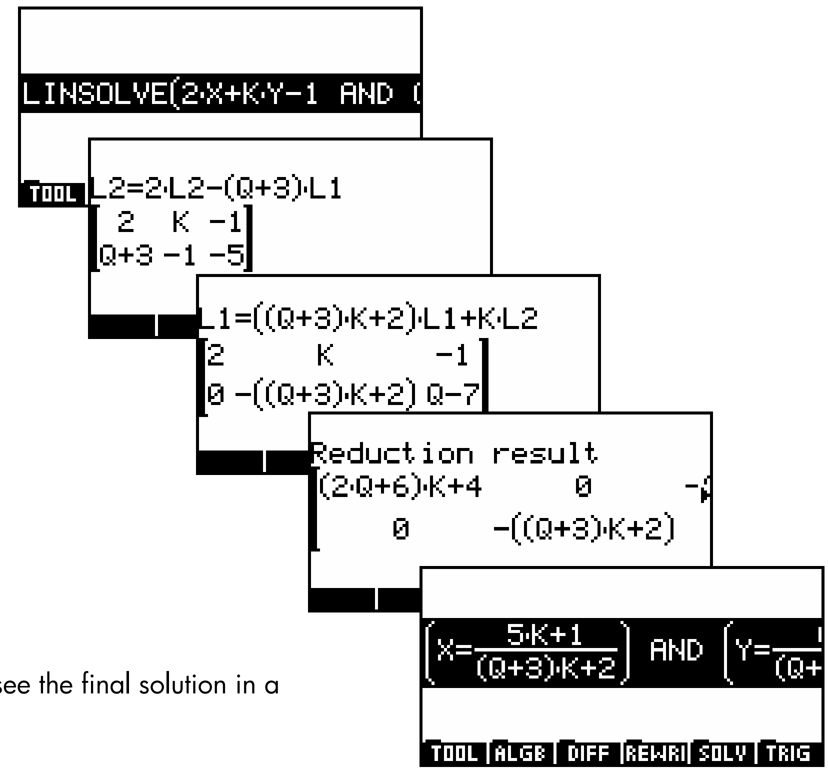
The **LINSOLVE** function can also be used to solve problems with symbolic coefficients such as the one below.

Solve the system of equations: $$\begin{bmatrix} 2 & k \\ q+3 & -1 \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 5 \end{bmatrix}$$

The command is

LINSOLVE(2.X+K.Y-1 AND (Q+3).X-Y-5, X AND Y)

and it produces the results shown.



Press **VIEWS** to see the final solution in a scrollable format.

Note that "KY" above must be entered as K*Y using [×] or the CAS will interpret KY as the name of a variable. In the CAS variables can have more than one letter, as is discussed on page 328.

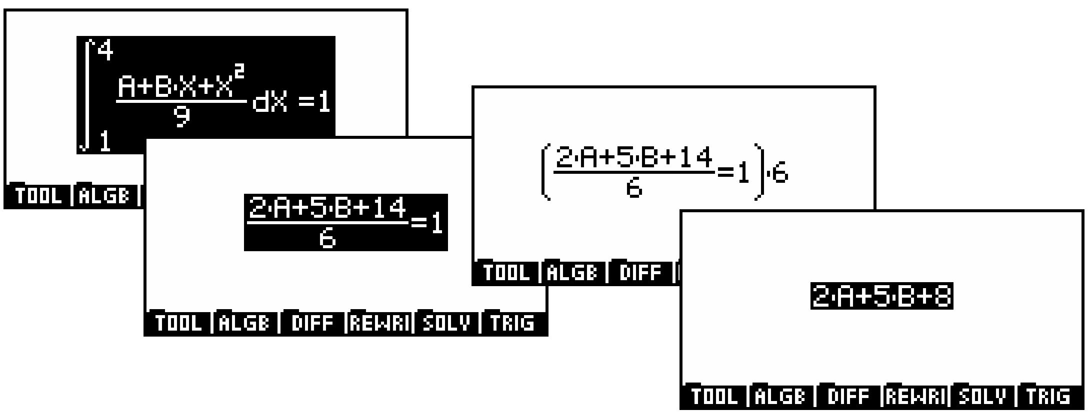
Example 7: Solving a simultaneous integration

A continuous random variable X, has a probability distribution function given by:

$$f(x) = \begin{cases} \dfrac{a + bx + x^2}{9} & for\, 1 \le x \le 4 \\ 0 & elsewhere \end{cases}$$

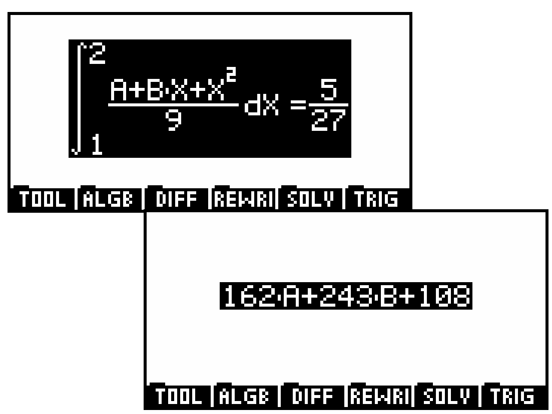
Given that $P(x \le 2) = \frac{5}{27}$, *find the values of a and b.*

From the fact that it is a probability distribution function we know that $\int_1^4 f(x)\, dx = 1$. We can use this to get the first expression in terms of a and b.

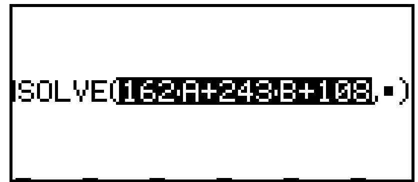


As can be seen above, the initial integration gives an equation involving a fraction. This can be simplified by multiplying both sides by 6, highlighting the entire equation first. Notice that when the final simplification is equal to zero, the calculator does not bother to include the '**=0**'. All expressions are assumed to be equal to zero unless otherwise specified.

The second probability tells us that $\int_1^2 f(x)\, dx = \frac{5}{27}$ and this gives the second of the pair of simultaneous equations in exactly the same fashion.



We can now use the **LINSOLVE** function to find A and B. While the second linear equation is still highlighted, fetch the **LINSOLVE** command from the SOLV menu. Then press ◄ and ▲ ▲ to highlight the linear expression again as shown right.

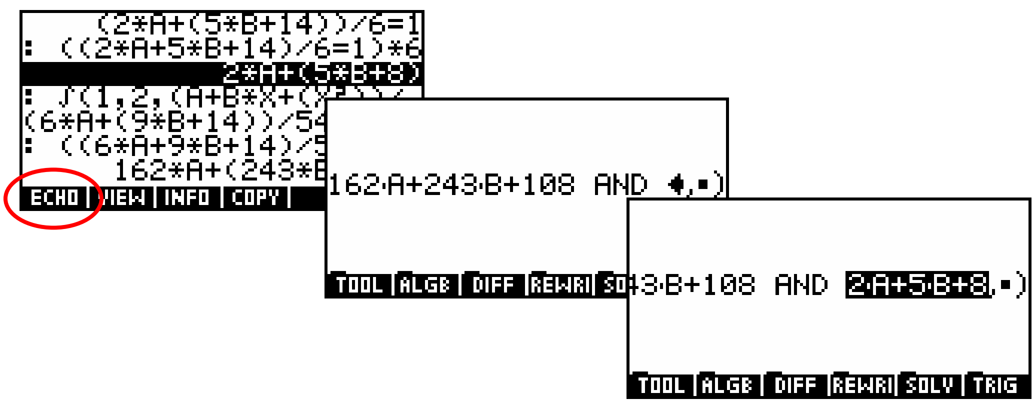


Now, while the entire expression is highlighted, as shown above, press **SHIFT (-)** to obtain the '**AND**'. The result should be as shown right.

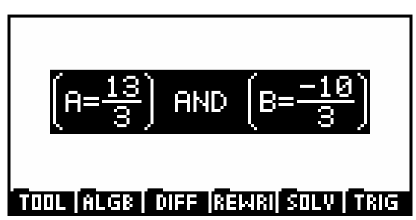
If yours looks like the version shown to the right then the error has been caused by not having the whole expression highlighted before pressing **AND**. Press **DEL** to remove the **AND**, then do it properly.

The other equation is still available from the CAS History. Begin by ensuring that no part of the current equation is highlighted and that the cursor is positioned where you want the new equation, immediately after the **AND**. If any portion of the expression in the editor is highlighted when you perform an ECHO then it will be replaced by the expression being echoed.

To fetch it, press **SYMB** and highlight the required equation. Pressing ECHO will echo it to the point of the cursor.



We can then finish off the **LINSOLVE** command by pressing right arrow to move to the second parameter position and then entering "**A AND B**". At that point we can press **ENTER** to obtain the values of **A** and **B**.



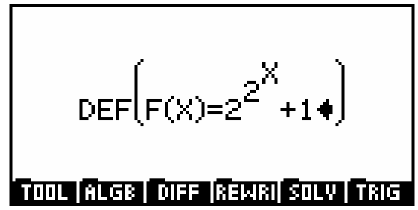
Example 8: Defining a user function

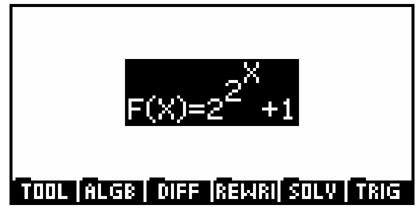
The **DEF** function allows you to define your own functions, which are then available for use. In the example below it has been used to define Fermat's prime function $f(x) = 2^{2^x} + 1$. Note that the sequence of keys is:

ALGB ▼ **ENTER ALPHA F (** X,T,θ

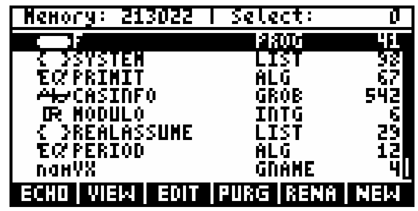
▲ ▲ **SHIFT = 2** x^y **2** x^y X,T,θ

▲ ▲ ▲ **+ 1 ENTER**



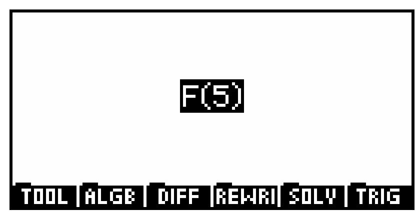


The CAS will then echo the function back to you and, if you press **VARS** to go to the **VARS** view, you will find that it is now a defined variable. Use the **VIEW** button at the bottom of the screen to see the definition of the variable. Press **ON** to exit the view.

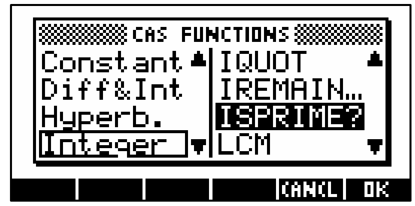


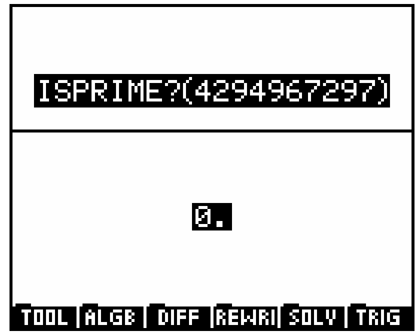
Although we used **X** as our variable here, there is no reason for this. We could just as easily have defined it as $f(K) = 2^{2^K} + 1$.

You can now call this function by simply typing, for example, **F(5)**.

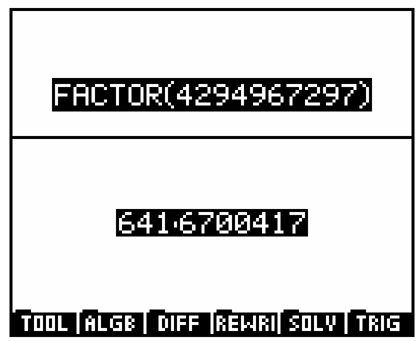


We can now test to see if this is a prime number by using the **ISPRIME?** function from the **MATH** menu. This is found in the Integer section of the CAS function list as shown right.





It returns a value of **0** (false) indicating that it is not a prime number.



Using the **FACTOR** function from the ALGB menu shows that its factors are 641 and 6700417.

Note: The **ISPRIME?** function gives correct results for integers up to 10^{14}. Beyond that level the results are highly probable to be correct but not guaranteed. They are obtained using "Rabin's Algorithm" and are called 'pseudo primes'.

Example 9: Investigation of a complex function

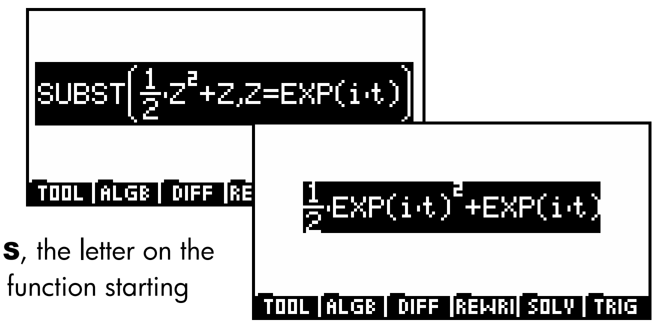
Rewrite the function $f(z) = \frac{1}{2}z^2 + z$ *in parametric form and graph it. Show that it is symmetrical about the x axis and evaluate* $f\left(\frac{\pi}{3}\right)$ *as an exact surd.*

i. The first step is to enter the function.

1 [÷] **2** [▲] [×] **ALPHA Z** [x²] [▶]
[+] **ALPHA Z**

ii. We now transform it into exponential form by using the **SUBST** function to replace z with $e^{i \cdot t}$.

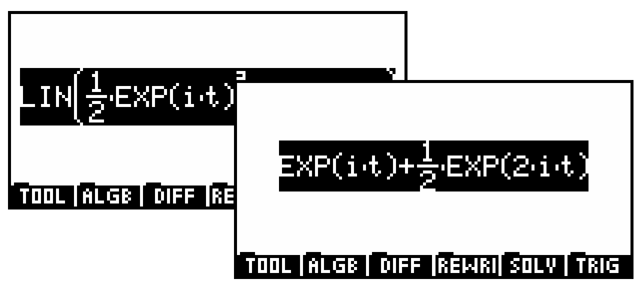
SHIFT [▲] **ALGB** [×] [×] **ENTER**
ALPHA Z SHIFT = SHIFT e^x
SHIFT i SHIFT ALPHA t
SHIFT [▲] **ENTER**



Note: The reason for pressing [×] within the menu is to jump to the first function starting with an **S**, the letter on the [×] key. Pressing it again jumps to the next function starting with **S**, which is **SUBST**.

iii. Next we linearize it…

REWRI [(] **ENTER ENTER ENTER**

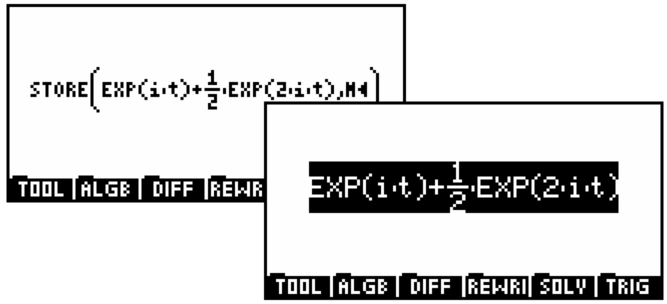


Note: 1. As in the previous case, the [(] is used to jump to the first function that starts with an **L**, the letter on [(].

2. The hp 40gs will probably ask if it should turn "Complex mode" on, assuming it is in its default configuration. One of the **ENTER**s is to tell it "Yes".

iv. And, having linearized it, we store it as a variable **M** in case we need to refer to it again.

ALGB × **ALPHA M ENTER**



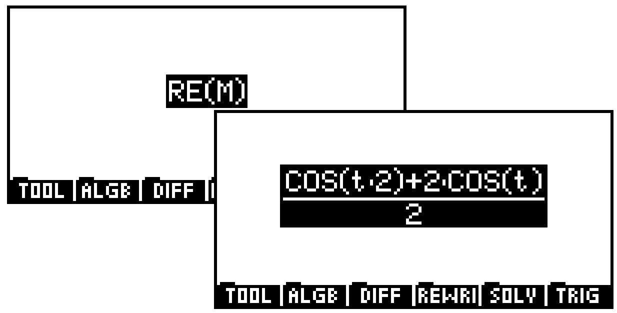
When the **STORE** command is executed the expression is echoed back to the screen.

Press **SHIFT ALPHA CLEAR** to clear the screen.

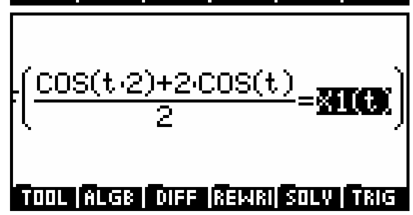
At this point, any reference to **M** will be equivalent to the expression shown right. Note that the screen image above is in small font purely to allow the entire expression to be seen. Your screen will not be unless you've selected small font earlier.

v. We now separate the real and imaginary parts of **M** into separate functions, storing them as **X1(t)** and **Y1(t)**.

ALPHA M ↑ **MATH** ↓ → **9**
ENTER ENTER

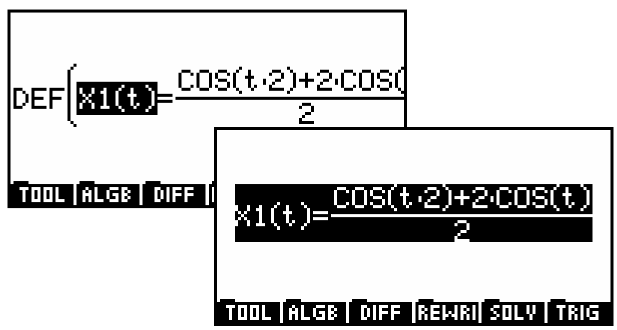


ALGB ↓ **ENTER** ↓ **SHIFT =**
ALPHA X 1 (SHIFT ALPHA T ↑ ↑



The parameter order must now be changed before the function is applied…

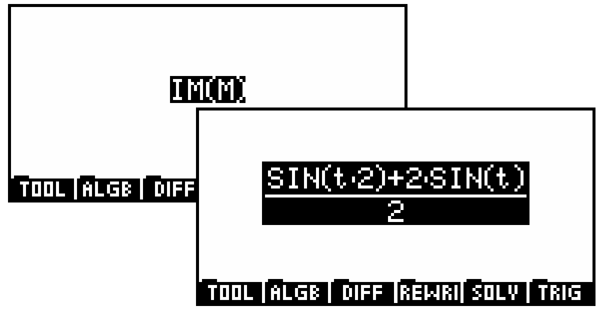
SHIFT ← **SHIFT** ↑ **ENTER**



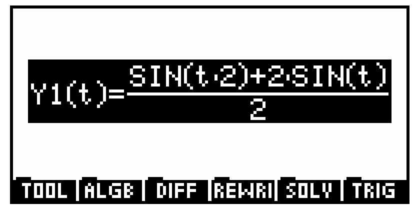
vi. Clear the current contents of the screen using **SHIFT ALPHA CLEAR**. Then perform the same definition assignment for **Y1(t)** as the imaginary part of **M**.

ALPHA M ▲ MATH ▼ ▶ [log]
ENTER ENTER

Note: As before, the [log] button jumps to the first function with that letter (**L**), in this case **IM**.



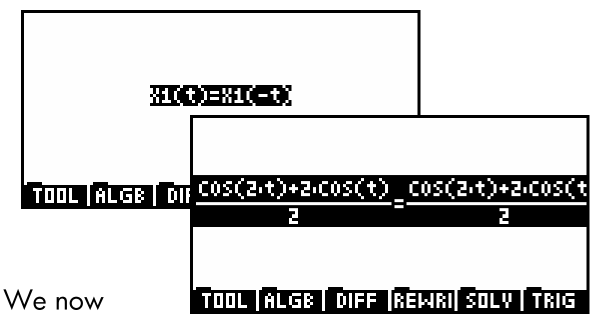
ALGB ▼ ENTER ▼ SHIFT =
ALPHA Y 1 (ALPHA SHIFT T ▲ ▲
SHIFT ◀ SHIFT ▲ ENTER



vii. In order to show that the function is symmetrical about the x axis we need to show that **(X1(-t),Y1(-t))** is equivalent algebraically to **(X1(t), -Y1(t))**. Purely for convenience we will display **X1(t)=X1(-t)** and evaluate each side. The = is not used to solve anything but just to display both at once for comparison so that it can be seen whether or not they are equal as required. To aid in this, we will change to the small font, clearing the screen first.

SHIFT ALPHA CLEAR TOOL ▼ ▼

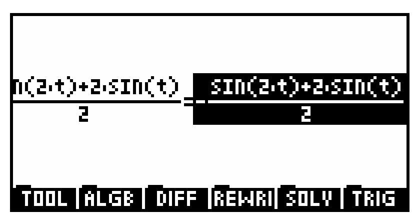
ALPHA X 1 (SHIFT ALPHA T ▲ ▲
SHIFT = ALPHA X 1 (SHIFT ALPHA T (-)
SHIFT ▲ ENTER



As can be seen, the two are algebraically equivalent. We now check the same equality for **Y1**, again clearing the screen first.

SHIFT ALPHA CLEAR

ALPHA Y 1 (SHIFT ALPHA T ▲ ▲
SHIFT = ALPHA Y 1 (SHIFT ALPHA T (-)
SHIFT ▲ ENTER ▼ ▶

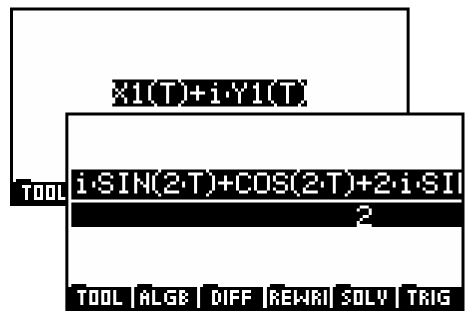


Again it is clear that the condition that **Y1(t)= -Y1(-t)** has been met. Therefore we can conclude that the graph is symmetrical about the x axis.

viii. We can see symmetry visually if the function is graphed and the aplet best suited to this is the Parametric aplet. When a function is sent to the Parametric aplet the real part is sent to the x and the imaginary part to the y component of the parametric equation selected.

> **Calculator Tip**
> Although **X1** and **Y1** were chosen as names for the functions defined earlier this has no relevance for the pasting into the Parametric aplet. They could just as easily have been defined as **Fred(t)** and **Jim(t)**. However, it is suggested that you not define a function as simply **X(t)**. This is because **X** is defined as the current variable by default and this means that the contents of your function definition will be purged as soon as you perform any operation that uses the current variable.

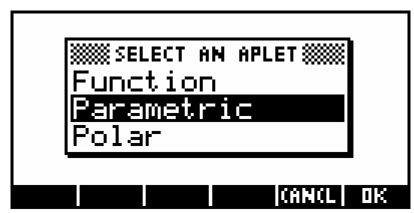
On the screen, define an expression $X1(T)+i\cdot Y1(T)$ which can be sent to the aplet, evaluating it and changing back to large font too.



SHIFT ▲ SHIFT ALPHA CLEAR TOOL ▼ ▼
ALPHA X 1 (ALPHA T ▲ ▲ + SHIFT 1
× ALPHA Y 1 (ALPHA T SHIFT ▲ ENTER

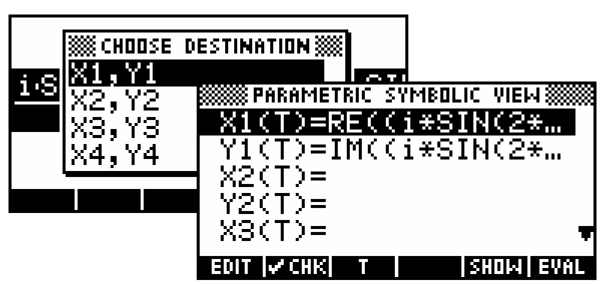
Now store to the aplet using the **PLOT** button on the keyboard, choosing **X1,Y1** in the Parametric aplet.

PLOT ▼ ENTER ENTER

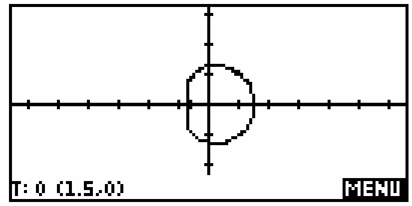


Note: The reason that we used capital **T** instead of lower case **t** here is because that is the variable used by the Parametric aplet.

We now change out of the CAS and activate the Parametric aplet in the usual fashion. When you do you should see the screen shown right. Each of these expressions needs to be simplified before graphing or it will significantly slow down the process. Highlight each in turn and press the **EVAL** screen button. Then press the **✓CHK** button.

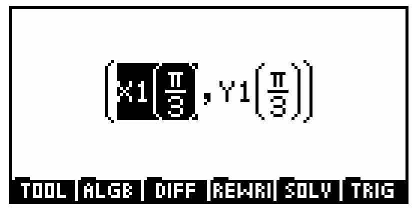


One additional step is required. For some reason the Parametric aplet doesn't seem to properly accept the functions. If you press **PLOT** now you will receive the "Undefined Name" error. The trick is to highlight each function in turn and press **EDIT**. Don't make any changes, just press **ENTER** or **OK**. This seems to make the calculator accept that it is a valid function, since it has had the chance to examine the function as if you had typed it in yourself. Pressing **PLOT** at this point will produce the screen shown right. The symmetry we predicted earlier is now visible. This may not be the best possible view and you may wish to use **PLOT SETUP** to change this.



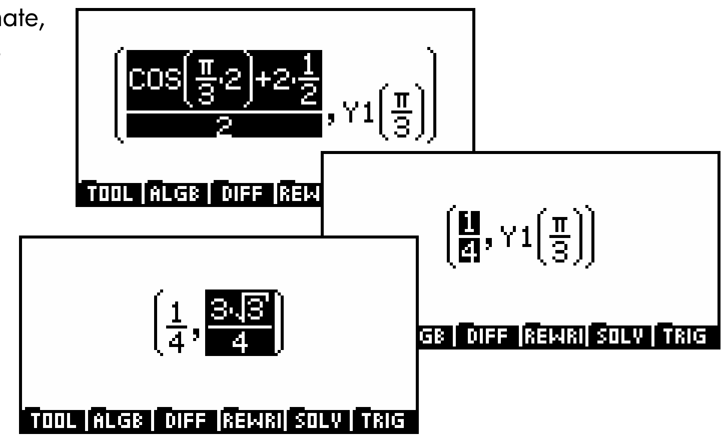
ix. The final step is to accurately evaluate $f\left(\frac{\pi}{3}\right)$. To do this, return to the CAS and enter the following:

ALPHA X 1 (SHIFT 3 [÷] **3 SHIFT** [▲] [,]
ALPHA Y 1 (SHIFT 3 [÷] **3 SHIFT** [▲] [▼]
(Notice the brackets appear when you press [,].)



We can now evaluate the left hand coordinate, then the other. The first enter substitutes the value, the second gives the answer.

ENTER ENTER [▶] **ENTER ENTER**

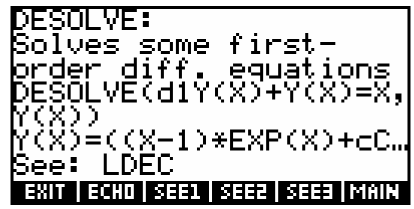


Example 10: First order linear differential equation

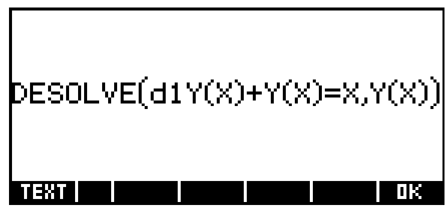
In order to illustrate the use of the CAS help pages discussed on page 361 we will the example provided in them rather than making one up. The functions available for solving differential equations are **DESOLVE** and **LDEC**.

Begin by pressing **SHIFT SYNTAX** to open the help menu and scroll down to the **DESOLVE** function as shown right. Pressing [X,T,θ] will jump immediately to the 'D's.

Pressing **ENTER** will display the screen shown right. Notice the reference to **LDEC** at the bottom of the screen. Pressing **SEE1** will jump to the screen for **LDEC**.



Press to paste the example into the CAS editing screen. The screen shot right has been obtained by pressing **VIEWS** and cutting and pasting in a Paint program to obtain a wider result. The '**d1Y(X)**' is used to represent the first derivative.



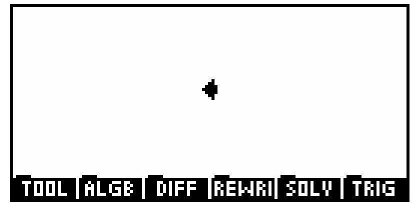
Highlighting this and pressing **ENTER** will give the result shown right. Note the use of '**cC0**' to represent a constant.

This is equivalent to $y = \dfrac{(x-1)e^x + c}{e^x}$. The CAS unfortunately uses **EXP(X)** to represent e^x, although it will understand the use of e^x when entering an expression.

The CAS menus

There are a variety of different places that functions are stored, often overlapping for greater convenience.

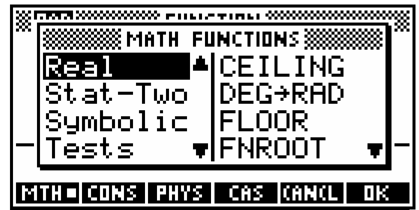
The Screen menus



On the main screen of the CAS you will see six labels of TOOL, ALGB, DIFF, REWRI, SOLV, and TRIG. The TOOL menu is discussed beginning on page 335. The others each pop up a menu of functions that are the most commonly used ones in each category but there are others that you can access via the MATH button or the **SHIFT CMDS** menu. The screen menu functions are listed in their categories below:

Algb.	*Diff.*	*Rewri.*	*Solv.*	*Trig.*
COLLECT	**DERIV**	**DISTRIB**	**DESOLVE**	**ACOS2S**
DEF	**DERVX**	**EPSX0**	**ISOLATE**	**ASIN2C**
EXPAND	**DIVPC**	**EXPLN**	**LDEC**	**ASIN2T**
FACTOR	**FOURIER**	**EXP2POW**	**LINSOLVE**	**ATAN2S**
PARTFAC	**IBP**	**FDISTRIB**	**SOLVE**	**HALFTAN**
QUOTE	**INVX**	**LNCOLLECT**	**SOLVEVX**	**SINCOS**
STORE	**lim**	**POWEXPAND**		**TAN2CS2**
|	**PREVAL**	**SINCOS**		**TAN2SC**
SUBST	**RISCH**	**SIMPLIFY**		**TAN2SC2**
TEXPAND	**SERIES**	**XNUM**		**TCLLECT**
UNASSIGN	**SIGNTAB**	**XQ**		**TEXPAND**
	TABVAR			**TLIN**
	TAYLOR0			**TRIG**
	TRUNC			**TRIGCOS**
				TRIGSIN
				TRIGTAN

The MATH menu



Pressing the **MATH** button in the CAS has a different effect than in the **HOME** screen. In the **HOME** screen the result is as shown right. As was discussed on page 165, these commands are broken into broad groups of MTH (mathematical), CONS (constants) and CAS.

The default set is the main mathematical functions shown right because they are the ones most often used in the **HOME** view but there are other menus accessible via the keys at the bottom of the screen (see page 165). The CAS functions are included because many of them can be used in the **HOME** view as well as in the CAS.

In the CAS view, the default is naturally the set of CAS functions shown right. These are the same ones you see in **HOME** if you select them but you may notice that the other menus (MTH, CONS & PHYS) are not available because many of them don't work in the CAS or are replaced by different versions. The functions available in the **MATH** menu in the CAS are as follows:

Algebra	*Complex*	*Constant*	*Diff&Int*	*Hyperb.*	*Integer*	*Modular*
COLLECT	**i**	**e**	**DERIV**	**ACOSH**	**DIVIS**	**ADDTMOD**
DEF	**ABS**	**i**	**DERVX**	**ASINH**	**EULER**	**DIVMOD**
EXPAND	**ARG**	∞	**DIVPC**	**ATANH**	**FACTOR**	**EXPANDMOD**
FACTOR	**CONJ**	π	**FOURIER**	**COSH**	**GCD**	**FACTORMOD**
PARTFRAC	**DROITE**		**IBP**	**SINH**	**IDIV2**	**GCDMOD**
QUOTE	**IM**		**INVX**	**TANH**	**IEGCD**	**INVMOD**
STORE	**-**		**lim**		**IQUOT**	**MODSTO**
|	**RE**		**PREVAL**		**IREMAINDER**	**MULTMOD**
SUBST	**SIGN**		**RISCH**		**ISPRIME?**	**POWMOD**
TEXPAND			**SERIES**		**LCM**	**SUBTMOD**
UNASSIGN			**SIGNTAB**		**MOD**	
			TABVAR		**NEXTPRIME**	
			TAYLOR0		**PREVPRIME**	
			TRUNC			

Polynom.	*Real*	*Rewrite*	*Solve*	*Tests*	*Trig.*
EGCD	**CEILING**	**DISTRIB**	**DESOLVE**	**ASSUME**	**ACOS2S**
FACTOR	**FLOOR**	**EPSX0**	**ISOLATE**	**UNASSUME**	**ASIN2C**
GCD	**FRAC**	**EXPLN**	**LDEC**	**>**	**ASIN2T**
HERMITE	**INT**	**EXP2POW**	**LINSOLVE**	$\geq$	**ATAN2S**
LCM	**MAX**	**FDISTRIB**	**SOLVE**	**<**	**HALFTAN**
LEGENDRE	**MIN**	**LIN**	**SOLVEVX**	$\leq$	**SINCOS**
PARTFRAC		**LNCOLLECT**		**==**	**TAN2CS2**
PROPFRAC		**POWEXPAND**		$\neq$	**TAN2SC**
PTAYL		**SINCOS**		**AND**	**TAN2SC2**
QUOT		**SIMPLIFY**		**OR**	**TCOLLECT**
REMAINDER		**XNUM**		**NOT**	**TEXPAND**
TCHEBYCHEFF		**XQ**		**IFTE**	**TLIN**
					TRIG
					TRIGCOS
					TRIGSIN
					TRIGTAN

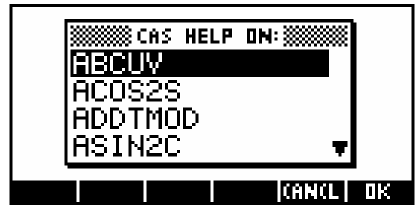
The CMDS menu

All of the functions listed in the table on the previous page are also available via the **SHIFT CMDS** menu where they are in alphabetical order rather than categorized. In addition to these there are also a number of other functions which appear only in that menu. They are:

ABCUX	**DIV2MOD**	**IABCUV**	**LAP**	**PLOTADD**	**SERIES**	**STURMAB**
CHINREM	**EXP2HYP**	**IBERNOULLI**	**LIMIT**	**PSI**	**SEVAL**	**TSIMP**
CYCLOTOMIC	**GAMMA**	**ICHINREM**	**PA2B2**	**Psi**	**SIGMA**	**VER**
DIV2	**HORNER**	**ILAP**	**PLOT**	**REORDER**	**SIGMAVX**	

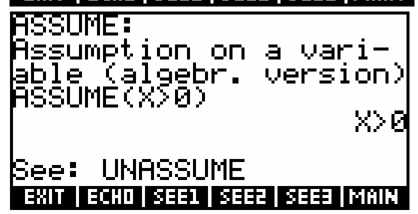
The abilities of the functions listed here and on the previous page will not be discussed in this text beyond the examples given on the previous pages. The manual supplied with your calculator is generally quite clear on their use and abilities.

On-line help



One of the most helpful features of the hp 40gs CAS is the on-line help provided by the **SYNTAX** button (**SHIFT 2**). Pressing **SYNTAX** will display the menu shown to the right. You can use the up or down arrow keys to scroll through this list but it is very extensive and it is far quicker to press the button corresponding to the first letter of the function on which you require information. For example, if you want help on the **NEXTPRIME** function then press the ÷ button (letter **N**) to jump to the first function beginning with **N**.

The help screen can be a little crowded, as you can see to the right, but generally contains an extremely useful summary.



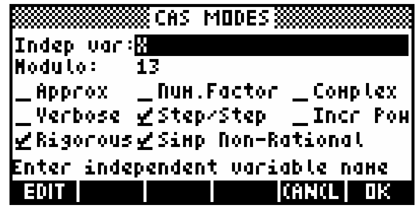
An example is always given and pressing **ECHO** will cause this example to be pasted into the editing screen, replacing whatever is currently highlighted in the editing screen.

There will usually be up to three cross-references to other functions. Pressing **SEE1**, **SEE2**, or **SEE3** will take you to the help screens for these functions.

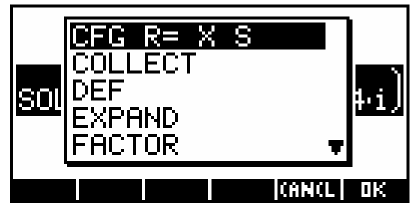
Pressing **EXIT** will take you back to the editing screen without any action, while pressing **MAIN** will take you back to the **SYNTAX** menu.

It is strongly suggested that you spend some valuable time going carefully through this collection of help pages. They will give you an excellent feel for the capabilities of the CAS.

Configuring the CAS



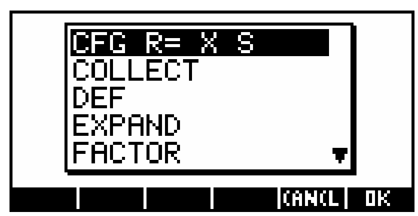
In most of the examples which precede this section it was assumed that the CAS was in its default settings. Two versions of the configuration screen are shown to the right.



These screens can be accessed within the CAS in a number of ways:

- from the **HOME** screen, pressing **SHIFT** **CAS**
- by pressing **SHIFT MODES** within the CAS
- by pressing **SHIFT SYMB** within the CAS
- via the **CFG** menu entry shown right. This appears as the first entry on most of the menus within the CAS. The configuration menu that this accesses is not the same screen as that shown above.

To ensure that the CAS is in default mode, enter the CAS and press **SHIFT MODES**. Now press **SHIFT CLEAR**. This will restore default settings.



Configuration of the CAS can be done in a number of ways and only an overview will be given here. One method is via the configuration line (**CFG**) at the top of each menu. The line shown right of **CFG R= X S** means that the calculator is set to **exact-real mode**, that **X** is the current variable, and you are working in **Step by step** mode.

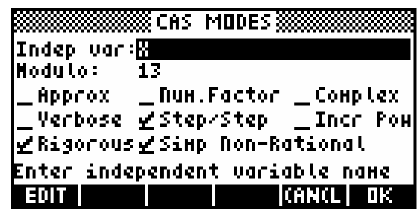
If you select this option of the menu then you will see the further menu shown to the right. At the top of this menu is more information about the current configuration.

The example to the right means that

R=	You are in exact-real mode
STEP	Step by step mode is selected
↓	Polynomials are written with their terms in descending order by exponent
X	X is the current variable
13	Modular calculations are carried out in Z/13Z (p = 13)
||	You are working in Rigorous mode (that is, using absolute values).

Below the title bar you can see the first section of a series of alternatives which let you manipulate the configuration. Most alternatives are toggles having only two values. For example, choosing **Complex** and pressing **ENTER** will cause the menu to momentarily disappear and then re-display with the new setting of **Real**. Pressing **ENTER** again will revert back to **Complex**. It is important to realize that the entry shown in the menu is the one that is NOT in use! For example, if it shows **Complex** then you are in **Real** mode and vice versa. Using the menu you can choose:

Quit config	(when you're finished making changes)
Complex	(or **Real**)
Approx	(or **Exact**)
Direct	(or **Step by Step** see following page)
1 + x + x²	(or **x² + x +1** ; governs how polynomials will display)
Sloppy	(or **Rigorous**, if you want to work in absolute values)
Num. factor	(or **Symb factor**)
Cmplx vars	(or **Real** vars if you want all symbolic variables to be assumed real)
English	(or **Français** if you want the help screens to be in French)
Default cfg	(which is **R= STEP ↓ X 13 ||**).



Pressing **CANCL** takes you out of the **CFG** menu (as does choosing **QUIT** from the menu and confirming it with **OK**) but does not discard any changes you have made.

The only difference between the two configuration menus shown right is that the lower one is harder to use and contains one option not available on the upper one – the ability to select French as the language in which the help screens are displayed.

The name of the current variable, as well as the value of the variable **MODULO**, can be changed by means of the **SHIFT MODES** view, or by using the **VARS** key. Many of the configurations addressed in the **CFG** menu are also referenced in these views. Press **ON** to exit the **VARS** configuration view.

The information supplied in the manual on the various configuration states is quoted below for reference together with comments and suggestions.

Approximate vs. Exact mode

When the approximate mode is selected, symbolic operations (for example, definite integrals, square roots, etc.), will be calculated numerically. When this mode is unselected, exact mode is active, hence symbolic operations will be calculated as algebraic expressions, whenever possible. [Default: unselected.]

Comment: Generally you will want to stay in Exact mode because this gives answers as surds, fractions of pi and so on. If you want a decimal result, simply highlight the expression you want converted and press **NUM**. Press **SHIFT NUM** to convert in the other direction. **SHIFT NUM** is equivalent to calling the **XQ** function.

Num. Factor mode

When the Num. Factor setting is selected, approximate roots are used when factoring. For example, is irreducible over the integers but has approximate roots over the reals. With Num. Factor set, the approximate roots are returned. [Default: unselected]

Comment: Some polynomials, particularly ones with complex coefficients, will not factorize using the **FACTOR** function without being in Num. Factor mode. See the example on page 346.

Complex vs. Real mode

When Complex mode is selected and an operation results in a complex number, the result will be shown in the form a + bi or in the form of an ordered pair (a,b). If Complex mode is not selected and an operation results in a complex number, you will be asked to switch to Complex mode. If you decline, the calculator will report an error or, in the case of factorizing, simply return the original expression. [Default: unselected.]

When in Complex mode, the CAS is able to perform a wider range of operations than in non-complex (or real) mode, but it will also be considerably slower. Thus, it is recommended that you don't select Complex mode unless requested by the calculator in the performance of a particular operation.

Comment: That says it all really. You need to be concerned with this if you are finding roots of polynomials. If Complex mode is not selected then only the real roots will be returned.

Verbose vs. nonverbose mode

When verbose mode is selected, certain calculus applications are provided with comment lines in the main display. The comment lines will appear in the top lines of the display, but only while the operation is being calculated. [Default: unselected.]

Comment: I've never found the comments to be particularly helpful. The methodology and algorithms that the CAS uses internally are not those that normal mathematicians use in many cases and so Verbose mode gives little useful information.

Step-by-step mode

When Step/Step is selected, certain operations will be shown one step at a time in the display. You press to show each step in turn. [Default: selected.]

Comment: This can be useful for functions such as **INTVX** and **LINSOLVE** (see earlier examples) where they give important information on how the problem was solved.
Try **INTVX(SIN(X)²/COS(X))** for an illustration. The information is stored in the **VARS** screen in the **CASINFO** variable and can be viewed again there if desired.

Increasing-powers mode

When Increasing-powers mode is selected, polynomials will be listed so that the terms will have increasing powers of the independent variable (which is the opposite to how polynomials are normally written). [Default: unselected.]

Comment: Who cares really?

Rigorous setting

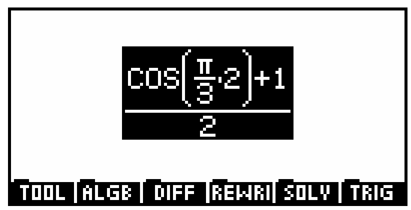
When Rigorous mode is selected, any algebraic expression of the form |X|, i.e., the absolute value of X, is not simplified to X. [Default: selected.]

Comment: Your choice here really depends entirely on the problem at the time.

Simplify non-rational setting

When Simplify non-rational mode is selected, non-rational expressions will be automatically simplified. [Default: selected.]

Comment: When this mode is not selected an expression such as the one shown right will not be simplified. When selected, it simplifies to ¼. I suggest that this is generally more desirable!



Calculator Tip

Changing the configuration can have a profound effect on the behavior of the CAS. Before you play with these configurations too extensively it is suggested that you read carefully the relevant information in the manual and/or the information in Renée de Graeve's book mentioned in the introduction to this appendix on page 324.

Remember that you can always return to the default configuration simply by going into the configuration view (**SHIFT MODES**) and pressing **SHIFT CLEAR**.

Tips & Tricks - CAS

- In CAS, angles are always expressed in radians and no other setting is possible. When you are the calculator **HOME** screen, you can use the **MODES** view to change this default but this does not affect the CAS.

- **Step by step** mode might appear to be quite useful for students but is quite limited in what it actually displays. Those hoping for the CAS to show complete working such as that required by a teacher will be disappointed. Choose **Step by step** mode to view some details of the calculations, displayed on the screen. After each step you are requested to press **OK** to proceed to the next stage. The alternative is **Direct mode**, in which only the result is displayed.

- Incredible as it may seem, the CAS is capable of infinite precision when manipulating integers (memory permitting). To see this in action try evaluating 200 factorial and then press **VIEWS**. You can now scroll through all 375 digits of the result!

- Don't use a summation variable of lower-case i. This is assumed by the CAS to be the unit imaginary value - the positive root of $x^2+1=0$.

- Although you can use the integration symbol provided on the keyboard it has disadvantages outlined on page 74. Use the **INTVX** function instead. See the example on page 339.

- The **COLLECT** function referred to earlier will factorize over the set of integers. For example, **COLLECT(x²-4)** will result in **(x+2)(x-2)**, whereas **COLLECT(X²+4)** will result in **X²+4** back again.

  On the other hand the **FACTOR** function will factorize over the irrational and complex sets too. Entering **FACTOR(X²+2)** will result in the expression $\left(x+\sqrt{2}i\right)\left(x-\sqrt{2}i\right)$ in complex mode.

- The infinity symbol can be found in the **Constants** section of the **MATH** menu but can also be obtained by pressing **SHIFT 0**. Pressing **(-)** first will produce -∞, while pressing **(-)** twice will give +∞. These are often needed for use in the **LIMITS** function.

  For example, evaluating **LIMIT(** $\sqrt{x+\sqrt{x+\sqrt{x}}}-\sqrt{x}$ **,+∞)** will give ½ (after a *very* long wait).

  As with the infinity symbol there is also a shortcut for the symbol i. Just press **SHIFT 1**. See page 333 for more keyboard shortcuts.