The TI-89 and how to use some of it for algebra, calculus, graphing, and a little bit of vector/matrix algebra

Prof. Lee Townsend, CETA, University of Hartford

Table of Contents

Topic	Page
Introduction	2
TI-89 picture, MODE	3
References	6
Setting up the calculator and creating a program	7
Menus	
Tools (F1)	9
Algebra menu (F2)	9
Calculus menu (F3)	13
Other menu (F4)	15
Clean Up menu (F6)	17
Apps	
Polynomial Root Finder	18
Simultaneous Eqn Solver	19
Doing Algebra	20
Graphing	20
Function graphing	21
Parametric graphing	24
Differential Equation graphing	25
Polar graphing	26
Trig	26
Vectors and Matrices	
Vectors	26
Matrices	28
Tables	30
Complex numbers	30
Assign keys 1-9 to applications	31
Unit creation and conversion	31
Bases	32
Inequalities	32
Statistics and regression	33
Tables of Contents for two TI books ($$)	36
Tables of Contents for MEPro, EEPro, EE200 (free)	36

Last modified January 27, 2006

Introduction

Since so many students are using the TI-89 yet don't know much about it, I have assembled this document, which contains what I have learned about this calculator in teaching math and physics along with significant self-study. This document is intended to present examples of the different functions. Some experience with graphing calculators, such as the TI-83 or TI-86, is assumed. The question is "how does the TI-89 do it?" You will find some of the answers in this document When all else fails, read the manual. The TI-89 is a computer algebra system (CAS) that is capable of interchanging information with the PC CAS software, *Derive*, available from TI (http://education.ti.com/us/product/software/derive/features/features.html). The TI-89 is programmable. Many application source files as well as executables are available from TICALC.org, http://www.ticalc.org/. The guidebooks for all the TI calculators are available in PDF format from the TI website, http://education.ti.com/us/product/main.html. More details on both sites are given in the reference section below. Please note that I am not an expert on the TI-89; I am just presenting what I use, how I do it, and where to find it.

I have written this document primarily using the technique of presenting screen shots resulting from some action. I do not give individual keystrokes. I assume you can read the screen to see what I typed. Be sure to look at the status line at the bottom of each screen to determine how my calculator is set up for the action under consideration. When giving a string of menu or button presses, I separate them by a /, e.g. MATH/Matrix/L:Vector ops. I present first the menus in the toolbar, *F1* through *F6*. After this information come sections on specific techniques such as graphing, solving simultaneous equations, regression analysis, etc. Finally, I give the table of contents for three applications and two books available from the TI website.

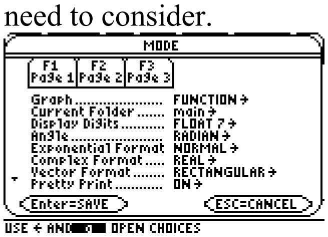
Where did I get my information? Most of it came from reading the printed guidebook that came with the TI-89 Titanium up to Appendix A and doing all the exercises. I actually found this guidebook to be a readable document. Some of my students have also taught me several uses of the commands.

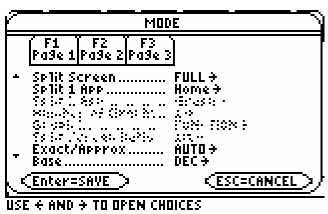
The TI-89 Calculator

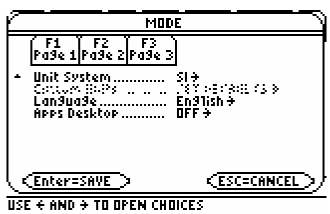
The calculator image is the *VTI.exe* TI-89 default skin

MODE

Set up the overall calculator settings by pressing the MODE button. There are three screens you need to consider.







A few general notes:

The calculator: Titanium or not?

The TI-89 Titanium has three times the memory of the first edition and communicates with desktops by USB instead of serial port. Otherwise, as far as I know, they are functionally the same. The OS for the original TI-89 stops at version 2.09. The Titanium starts with 3.0 and will be upgradeable. TI calculators can connect to each other for program/data exchange by the supplied cable.

The interface:

On the TI-83, 86, etc., you push a button and, typically, a command is executed.
On the TI-89 you typically construct a command on the entry line at the bottom of the screen by pressing buttons. Then, press ENTER to execute the command you constructed.

From MODE: Be sure to leave Pretty Print on because then when you enter an equation, it shows up in the history area looking typeset. It is therefore fairly easy to check that you typed what you meant to type. I find most students miss a parenthesis or two so having this feature available on the calculator is quite helpful.

You will also need to choose how you want your numbers to look – *approximate* (decimal) or *exact* ($\sqrt{\ }$, $\tan^{-1}$, π, etc. in the result). Choosing *auto* means that, when possible, the results will be presented in *exact* form (not always a pretty sight), or *approximate* if *exact* is not possible. To force *approximate*, be sure to put a decimal point on some number.

Screen Contrast:

♦ + and ♦ – make the screen darker and lighter. The ♦ symbol means "press the *Diamond* button." It is just below the 2nd button.

Using previous expressions:

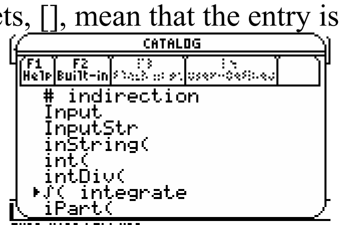
Scroll up and down in the history area. When the expression you wish to use on the entry line is highlighted, press the ENTER button. The highlighted expression is then placed on the entry line at the current cursor location.

Memory Management:

2nd – (VAR-LINK) places you in the right area. Read the Titanium Guidebook (the printed guidebook) for details. That's what I do.
2nd 6 (MEM) displays how memory is being used.

A few notes on using some features:

1) ***Catalog*** lists all the functions as well as expected input at the bottom of the screen. The brackets, [], mean that the entry is optional.



See F3/2 – $\int$(below for examples of the use of the integrate function with its input options.

To get to *and*, for example, in the catalog, press the CATALOG button, then press the = button to get to the beginning of the *a*'s. Cursor down once to get *and*. Using the catalog is typically much faster than using the menu system if you know the name of the command you are looking for.

2) Some menu items are commands and some are text that will show up on your entry line.

3) Press the ON button to stop a calculation in progress.

4) 2^{nd} ← moves the cursor to the beginning of the entry line and 2^{nd}→ to the right end.

5) If you position your cursor in the middle of a line, the first *Clear* eliminates the text to the right of the cursor and the second *Clear* finishes clearing the entry line.

6) The HOME button brings you back to the main (HOME) screen. 2^{nd} HOME toggles the **custom menu**. You can create your own custom menus. See the guidebooks (references #1). If your menus look weird, you probably pressed 2^{nd} HOME. Press 2^{nd} HOME again to get back to the original menus. Now you know how to recover the default menus.

7) Menu selection – you can scroll down to a menu selection, scroll up to it (good for menu items near the bottom of the list), or just press the selection's menu number on the numeric key pad.

8) When a menu item has an arrow (►) to the right, press the right cursor arrow, →, to see a submenu.

9) "Internally, the TI-89 converts all entered trig values to radians, but it does not convert values for exponential, logarithmic, or hyperbolic functions." [p 565 Electronic User Guide] In other words, be very careful which angle mode your calculator is in. The current mode is shown in the status line.

10) Spend some time looking at all the options under the MATH (2^{nd} 5) menu. There are many options listed there that I do not cover in this document.

11) Typing literal characters: Keep pressing the ALPHA button. The calculator goes through a sequence of alpha-off, alpha-on for one character, then alpha-stay on. Watch the second entry in the status line. First there is nothing, then a lower case *a* then a highlighted *a*, respectively. The 2^{nd} key is an on-off toggle. 2^{nd}/ALPHA directly locks alpha mode. Look at the status line of the three screen shots below.

Upper case characters are created by pressing the up-arrow key just above the ALPHA key (not the up-arrow cursor key), followed by the ALPHA key. Note the *A*, versus *a*, in the status line.

REFERENCES

1) TI-89 Guidebooks and application software.
   To download them go to the TI web site:
   http://education.ti.com/us/product/main.html
   Choose: Graphing Calculators and Accessories
   Select TI-89 Titanium

   A)
   Choose: Guidebooks from the menu on the left.
   Select **Printed Guidebook** (the little one that comes with the Titanium)
   Filename from TI: *TI89TitaniumGuidebook_Part1_EN.pdf*
   Or **Electronic Guidebook** – the full guidebook for the TI 89
   Filename from TI: *TI89TitaniumGuidebook_Part2_EN.pdf*
   Then Choose your language. I choose English, hence the *EN* at the of the guidebook filenames.
   Then Download the PDF file.

   B)
   Choose: Apps from the menu on the left.
   This is the place where you can download all the applications, most of which are free, along with their guidebooks in PDF format.

2) TI Calc: http://www.ticalc.org/
   Select **Archives** to find "thousands of files for every TI graphing calculator model"
   Select **Programming** to find emulators and programming help

3) Brendan Kelly, Advanced Algebra with the TI-89, Brendan Kelly Publishing, 2000.

4) Voltmer and Yoder, Electrical Engineering Applications with the TI-89, 1999.
   http://education.ti.com/us/product/book/89eea.html
   "This book is written for electrical engineering students. It is a collection of examples that show how to solve many common electrical engineering problems using the TI-89." I find this book to be an excellent source of TI-89 information, not just information specific to electrical engineering. One chapter includes writing functions for gradient, curl, divergence, the Laplacian, etc.

5) Some of my students.

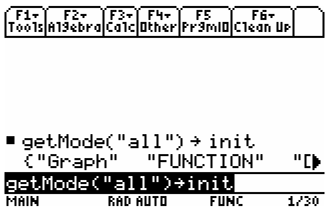
All screen shots in this document were taken using the original TI-89 ROM running OS 2.09 in *VTI.exe*. Unless otherwise indicated, I learned how to do everything that follows from the TI Guidebooks, reading the input information from CATALOG, and by following my instinct.

Acknowledgements. Thanks to Prof. Segal, Gale Townsend, and two of my students, Dave Johnson and Neil Patel, for proofing and editing this document. All errors are mine.

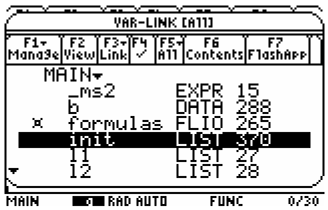
Setting up the Calculator

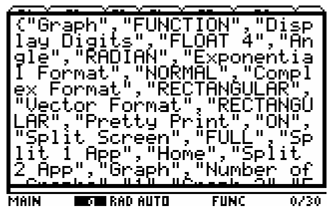
[4, §"Do This First"]

To store your current mode settings use CATALOG/getMode("all") STO► init

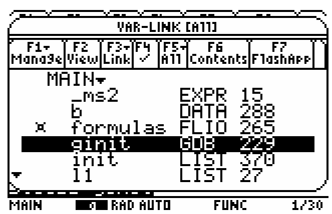


You can look at the beginning of what is stored: VAR-LINK/init/ F6 (Contents)





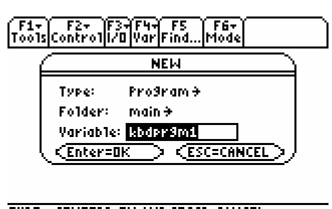
Also store the graphics settings and variables.   CATALOG/StoGDB ginit.



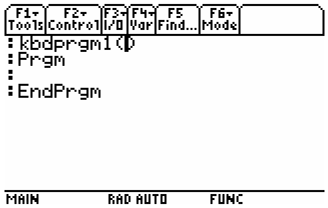
The file *ginit* is not a text file or a list so cannot be viewed.

To restore the MODE and graphics settings to those stored in *init* and *ginit*:

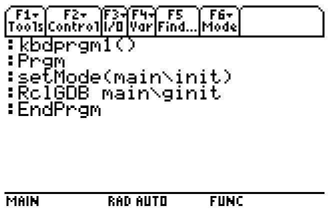
For ease of use when starting a new problem, Voltmer and Yoder [4, p3] recommend writing the following program that restores your calculator to a desired stored mode.   By naming the program *kbdprgm1* (Keyboard program #1), all you have to do is press ♦1 to run the program. To create this program, start from Apps/Program Editor



You are now presented with an empty program stub:



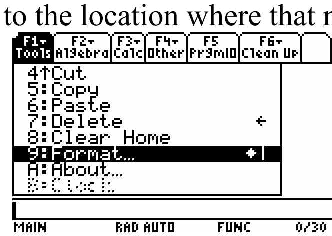
Create the program as given below. Once you have finished typing the program, press the HOME key to exit program mode.

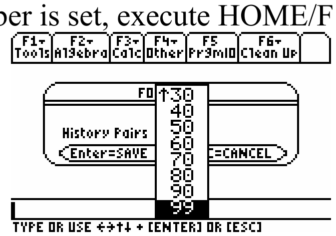


Voltmer and Yoder recommend that before you start a new problem session

1) Restore the mode settings stored in *init* and *ginit* by pressing ♦1
2) Run HOME/F6/2:NewProb to clear out the variables used in the last problem.
3) Create a new folder (HOME/F4/B:newFold) to use for the new problem. Note that by using different folders for different problems, it is much easier for you to look in the folder by using VAR-LINK/F2/FolderNameOfChoice. I have not done that and now my *main* folder is so full it is very tedious to find, look at, archive, lock or delete any of my variables (VAR-LINK/Manage).

Since NewProb clears the HOME screen, you can now save the steps of the current problem only into a text file with executable lines at the end of your problem session for later reference and execution (HOME/F1/2:Save Copy As..). Note that my calculator is set to remember 99 operations. To get to the location where that number is set, execute HOME/F1/9:Format.



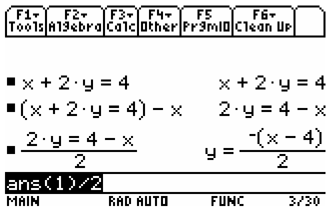


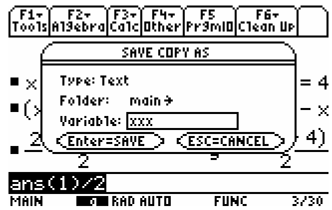
You might also consider writing a few more *kbdprgm* programs. Several that come to mind are programs to toggle between radian and degree modes, toggle between approx and auto modes, and conversion between radians and degrees. For example, the appropriate commands for setting degree mode is *setMode*("angle","degree"). After this command is run, to see that it worked, run *getMode*("angle")

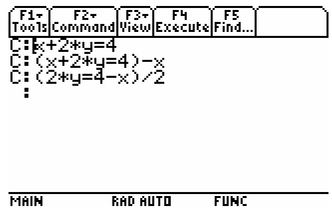
Menus with Examples

F1 – Tools

2 – **Save copy as**.
This command lets you save the history as an executable text file.







The text editor is an icon under APPS if you are using the Apps Desktop. I have turned off the icon desktop (MODE/F3/Apps Desktop/OFF) so my text editor shows up in a list when I press the APPS button. The file *xxx*, above, is a text file that can be edited. The C: means the text line is a command that can be executed Select the command, then press *F4* to execute it. The command is executed on the Home screen and then you are returned to the open text file. You can see what was executed by pressing the HOME button. This method is a nice way to save class notes. Clear the home screen before class. At the end of class save all your calculations so you can see what was done during class. Note that you can buy a full-sized keyboard from the TI web site so you can actually type notes into a text file.

3–>8 **Copy, Cut, Paste, Delete, Clear Home** behave as you would expect them to. Delete is the same as the backward arrow.

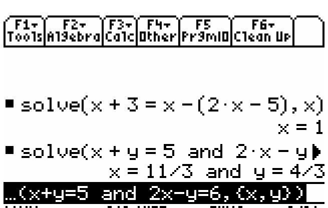
9 – **Format** lets you set how many calculations are saved in the history. I use 99 so I can save a large history file.

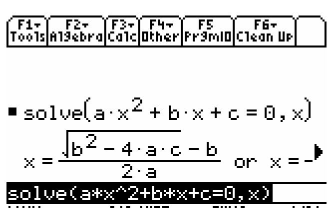
A – **About**: gives you the operating system (OS) version and calculator serial number.

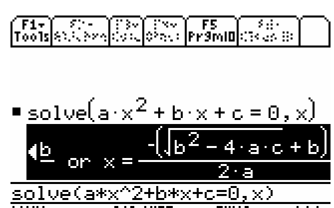
B – **Clock** (Titanium only): where you set the time and date. This information appears on the Apps Desktop.

F2 – Algebra

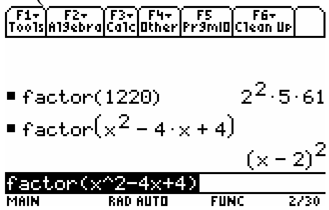
1 – **solve(**

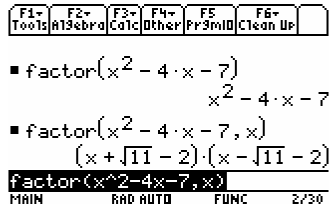


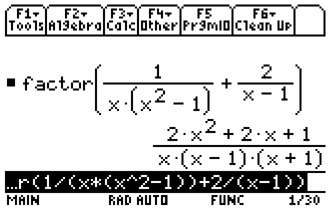




2 – **factor(**

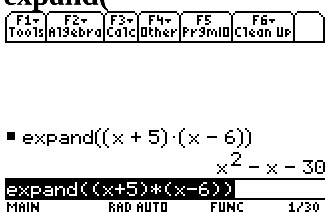


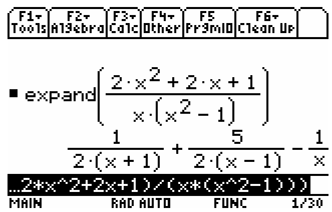


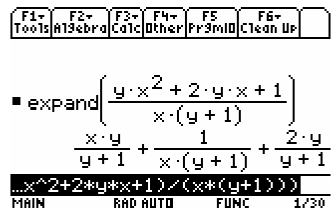


(I found this use of factor from a student on his exam.)

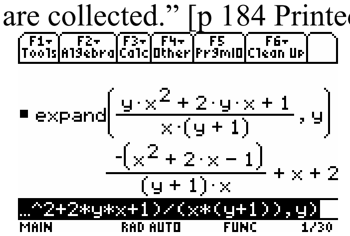
3 – expand(

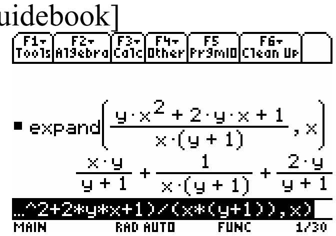




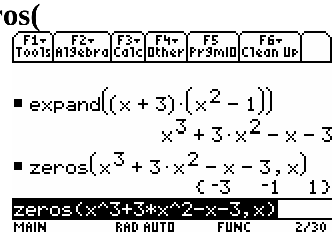


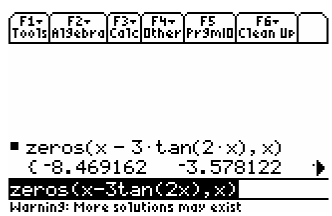
"expand(expression1,var**)** returns expression expanded with respect to var. Similar powers of var are collected." [p 184 Printed guidebook]

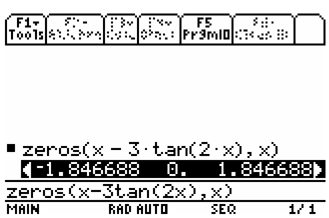


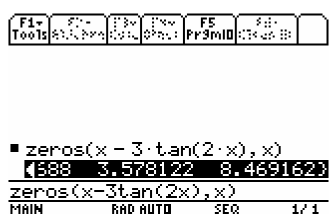


4 – zeros(

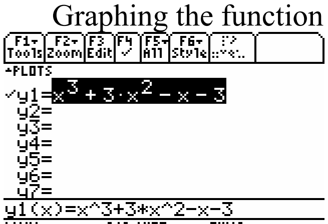


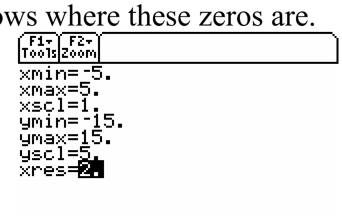


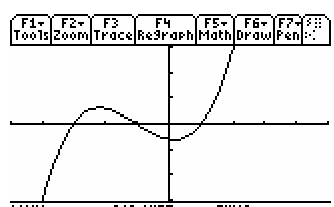




Graphing the function shows where these zeros are.

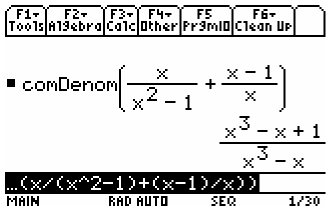




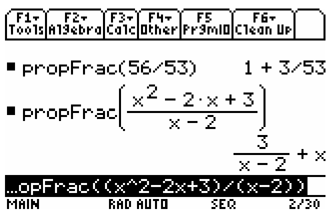


5 – approx() returns a decimal, independent of Exact/Approx mode. Alternatively, make sure to place a decimal in one of the numbers in your calculation to be assured of getting a decimal result. You can usually use ◆ ENTER to turn the previous result into a decimal as well.

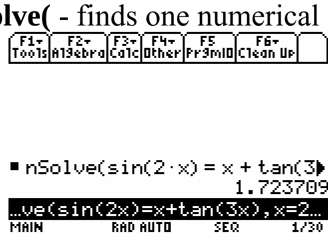
6 – comDenom("returns a reduced ratio of a fully expanded numerator over a fully expanded denominator." [p 161 Printed Guidebook]



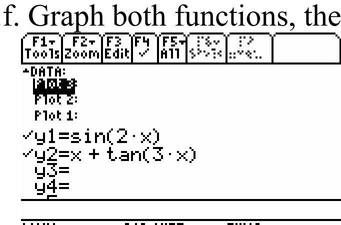
7 – propFac("returns the sum of proper ratios and a polynomial with respect to *var*. The degree of *var* in the denominator exceeds the degree of *var* in the numerator in each proper ratio." [p 221 Printed Guidebook]

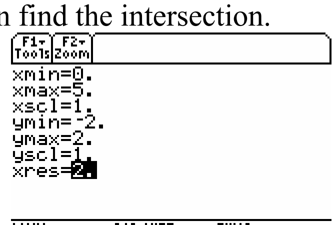


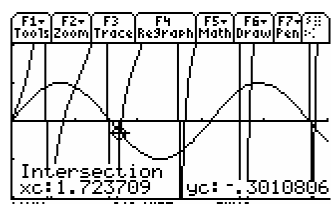
8 – nsolve(- finds one numerical answer (c.f. TI-86 SOLVER)



c.f. Graph both functions, then find the intersection.

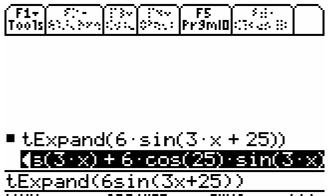


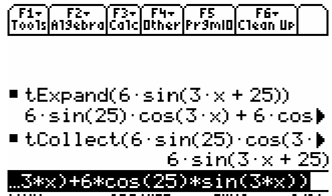


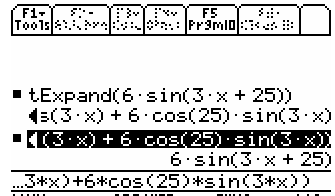


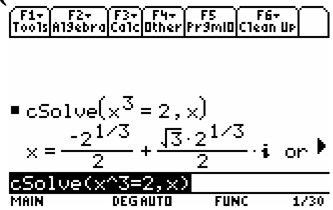
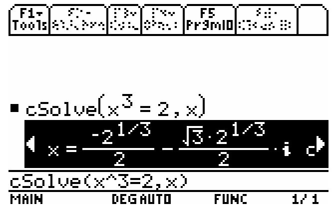
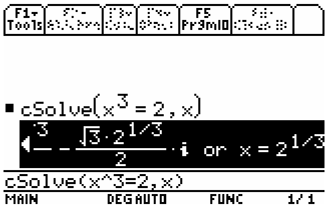
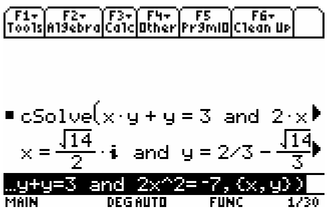
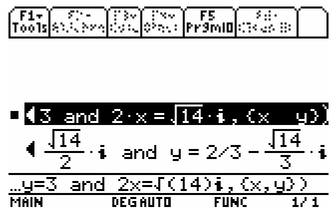
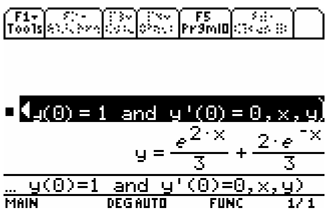
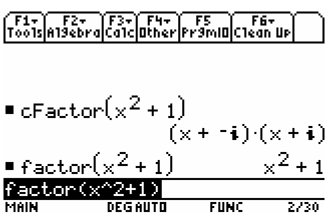
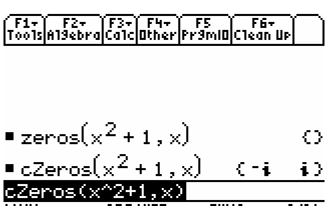
9 – Trig

tExpand(and **tCollect(** (gave odd results in Radian mode)



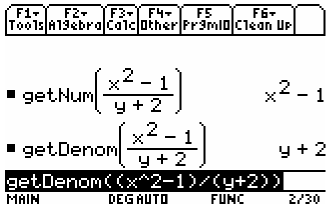


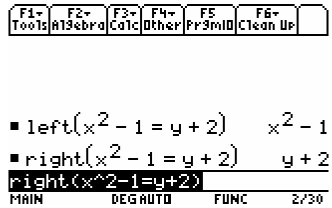


	A – Complex **cSolve(**      
	cFactor(
	cZeros(

B – Extract

getNum(and **getDenom(** and **left(** and **right(**

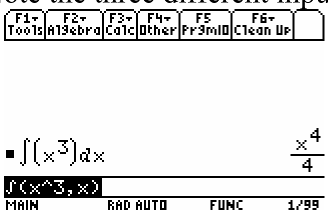


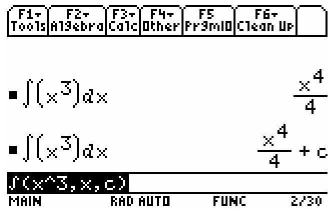


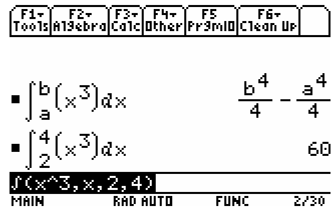
***F3* – Calc**

2 – ∫(Integrate

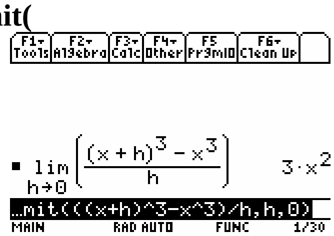
Note the three different input options**.**



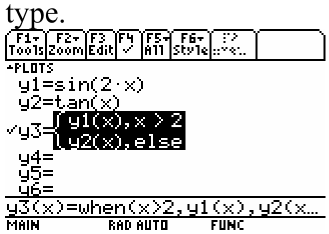


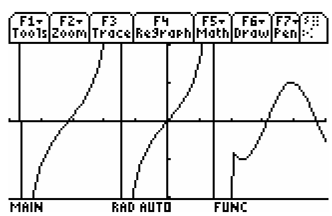


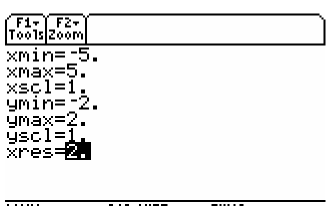
3 – limit(

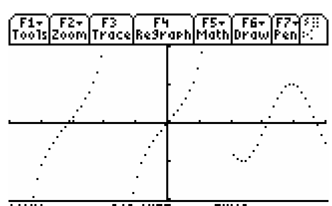


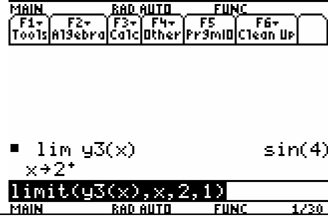
Use *when* for discontinuous functions; you can then take limits from the left and right of the discontinuity. This function was given me by a student while we were studying inequalities. The vertical lines in the graphs are artifacts of the graph being constructed by connecting points. They disappear when dots are used for the line type.

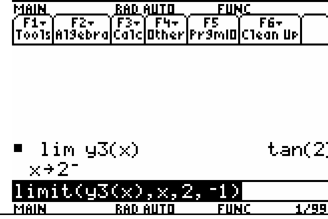




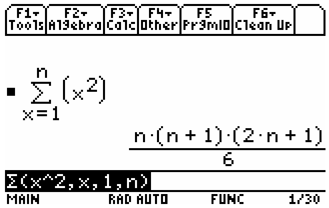




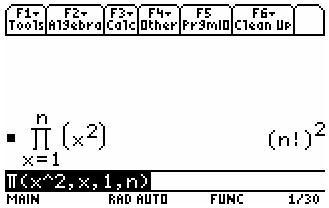




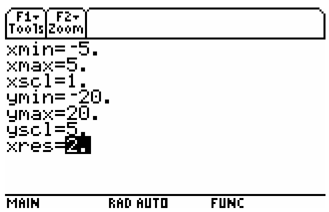
4 – $\sum($ sum

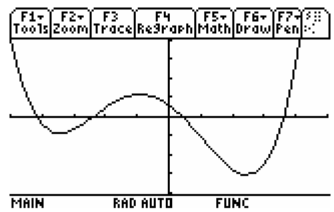


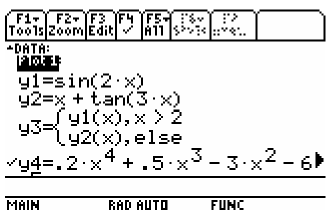
5 – $\Pi($ product

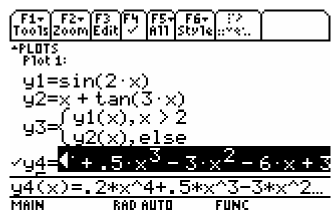


6 – fmin(and 7 – fmax(and the *Where* operator, |

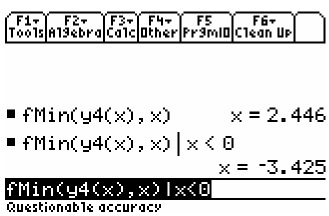


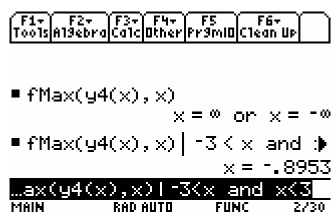






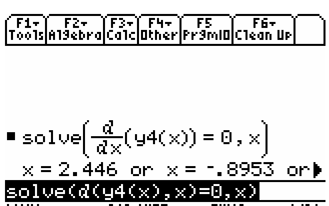
Note that graphs *y*1, *y*2, and *y*3 are deselected above so only *y*4 shows up on the graph.

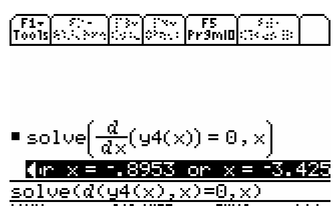




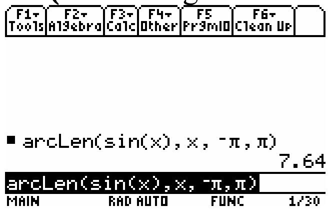
Note the use of the “where” operator |; e.g. find the minimum value of y4(x) where x<0.

Find the same result using SOLVE and some calculus – extrema occur when $\frac{dy}{dx} = 0$.

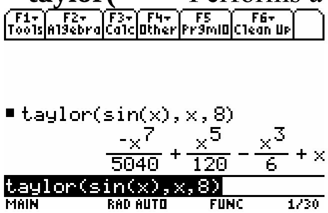




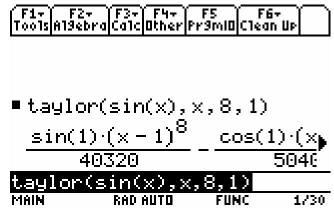
8 – arcLen(- the length of a line defined by a function.



9 – taylor(Performs a Taylor series.

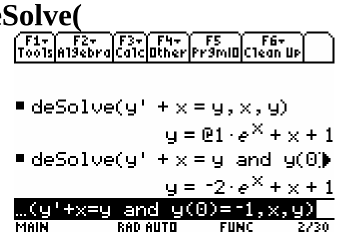


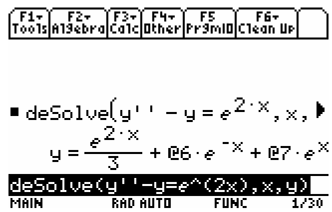
Perform the expansion about 1:

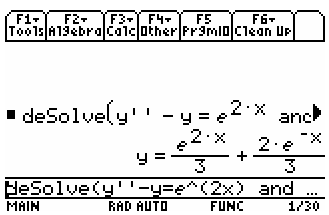


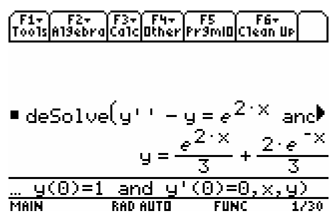
A – nDeriv(I don't use these.
B – nInt(

C – deSolve(





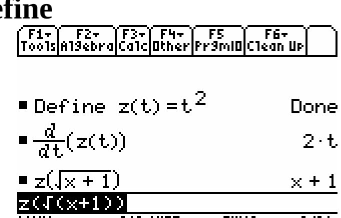




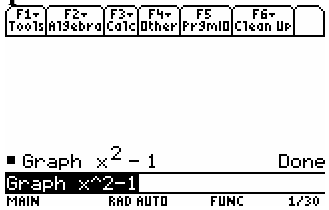
If you choose to solve for the constant @1, you can find the @ under 2nd + (the CHAR button)/Punctuation/9:

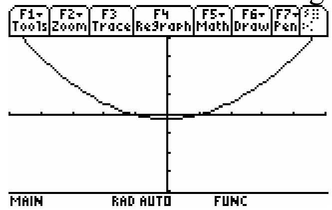
***F4* – Other**

1 – Define



2 – Graph – uses the current window. Making a **selection to copy**.

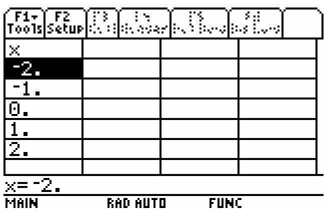


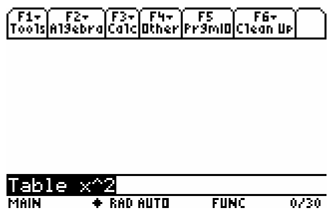


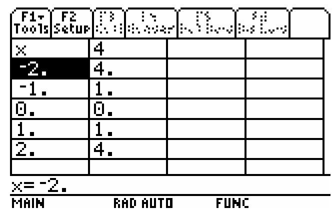
You can turn x^2-1 into a function yn(x)= x^2-1 by copy/paste. Select the x^2-1: Place the cursor at the right end of the expression by using 2nd →. Press the ↑ button (just left of the ESC button). Move the cursor to the left, continuing to press the ↑ button, over the x^2-1. Each character is selected as you cursor over it.

Copy the selection (*F1*/5 or ◆ ↑), go to the *y*= window, then paste it (*F1*/6 or ◆ ESC) into one of the *y*'s.

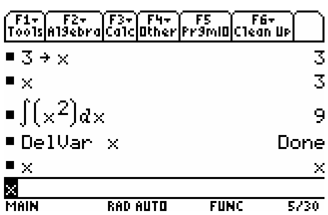
3 – Table (use Apps/Table to see the contents of your table.)

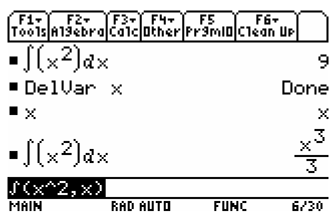






4 – Delvar what to do when you expect a variable but get a number instead.





5 – ClrGraph Clear the graph screen (used in programming).

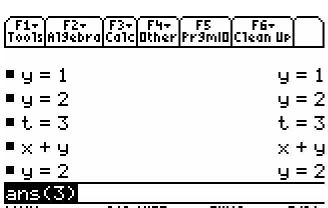
6 – ClrIO **"**Clear the TI-89 Program IO (input/output) screen." (used in programming) [p 741 Electronic Guidebook]

7 – FnOn/ 8 – FnOff

Select/Deselect all the functions defined for the current graph mode (*y*=).

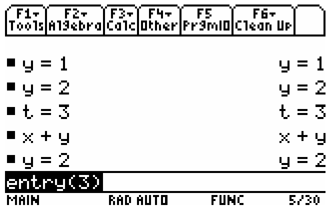
9 - Ans(

This command allows you to use a previous result. Use 2nd (-) for Ans(1). To go further up the history chain, use F4/Ans(.



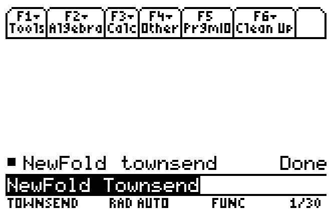
A – Entry(

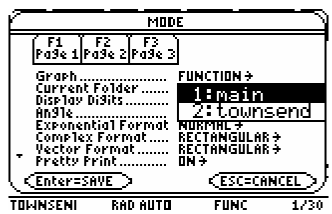
This command allows you to execute a previous command. Use 2nd ENTER to edit the previous calculation. To repeat a command further up the history chain, use F4/Entry(or just cursor up until the desired entry in the history area is selected, then press ENTER



B – NewFold

Create a new folder. Since the folder name is at the bottom of your screen, you can create a folder with your name so it shows. This idea comes from one of my students. Below, I create the *Townsend* folder. The folder name appears in the status line. I then use MODE/F1 to return the folder to *main* because that is the folder I am currently using for this document. Note that in the previous screen above, we were in the *main* folder. You could create folders for each class in which to store calculations and their results (F4/B:NewFold).





F6 – Clean Up

1 – Clear a-z

"Clears (deletes) all single-character variable names in the current folder, unless the variables are locked or archived" [p 178 Electronic Guidebook].

Note that temporary variables should be stored with single characters and those you want to keep with two or more characters. Then you use this command to delete all the temporary variables. Other variables can be deleted easily using *Delvar*, as described above.

2 – NewProb

"performs a variety of operations that let you begin a new problem from a cleared state without resetting the memory" [p 178 Electronic Guidebook]. According to Voltmer and Yoder [4, p 9], "It clears all single characters, unlocked variables; turns off all functions and stat plots; and clears all errors, graphs, tables, and the program I/O and Home screens."

3 – Restore custom default

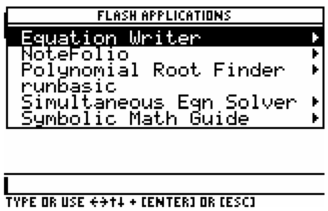
"If a custom menu other than the default is in effect,
this command lets you restore the default" [p 178 Electronic Guidebook].

Apps and FlashApps and Other Activities

Since I have turned off the *Apps Desktop*, my screen may look different from yours. If you turn on *Apps Desktop*, all entries in *Apps* appear as icons. I felt I had too many icons to wade through to get to the application I was looking for.

Pressing the *Apps* button gives

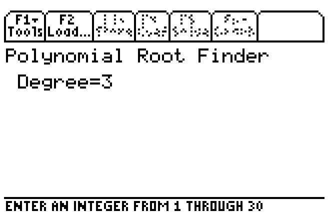
Selecting *FlashApps* gives

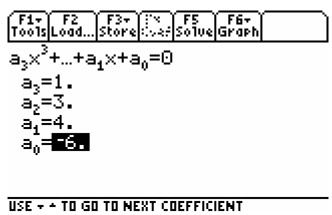


Your loaded flash apps may be different. If you do not have the *Polynomial Root Finder* and the *Simultaneous Eqn Solver*, be sure to get them from the TI web site.

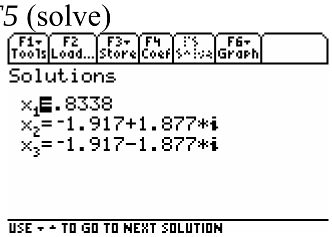
Polynomial Root Finder

Select NEW, then find the roots of $x^3 + 3x^2 + 4x - 6$.

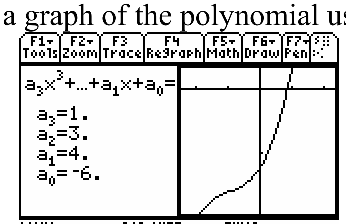




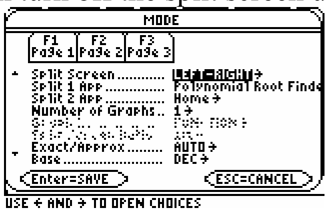
Press *F5* (solve)

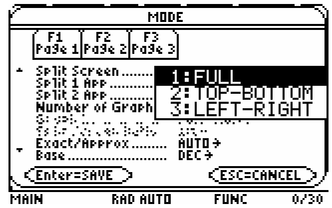


Pressing *F4* returns you to the coefficient entry screen. Using *F1*, you can store or load (or use *F2*) coefficients. Pressing *F3* presents you with the option of storing your equation in the *y* editor. Pressing *F6* allows you to show a graph of the polynomial using a split screen.



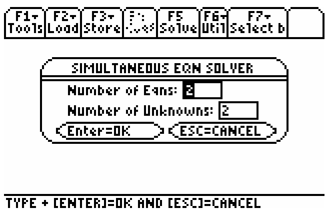
You can turn off the split screen using MODE/F2:

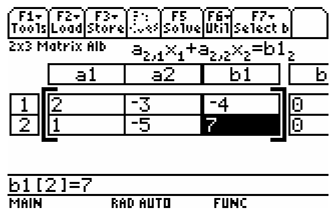




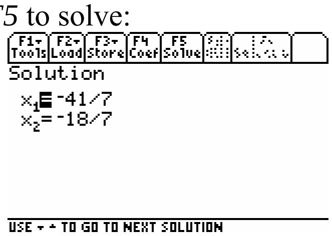
Simultaneous Eqn Solver

Select *New*, then solve $\begin{array}{l} 2x-3y=-4 \\ x-5y=7 \end{array}$





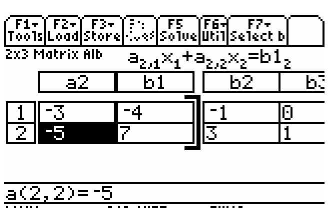
Press *F5* to solve:

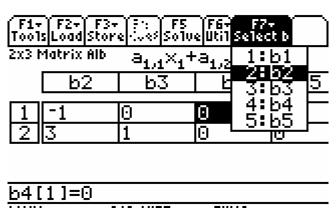


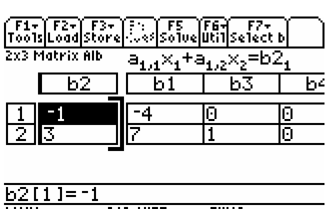
Press *F4* to return to the coefficient matrix to add more equations with different right-hand sides but the same left-hand side, such as solving for currents with different voltages but the same circuit otherwise. If you don't need to save the information, just overwrite the *b* information. However, if you do wish to keep the information, you will need to add more b's to the table. Now we wish to solve

$$\begin{array}{l} 2x-3y=-4 \\ x-5y=7 \end{array} \quad \text{and} \quad \begin{array}{l} 2x-3y=-1 \\ x-5y=3 \end{array} \quad \text{and} \quad \begin{array}{l} 2x-3y=0 \\ x-5y=1 \end{array}$$

Using the cursor (→) gets you to the entry area for the other equations. Use *F7* to choose which right-hand side you wish to use. Press *F5* to solve.

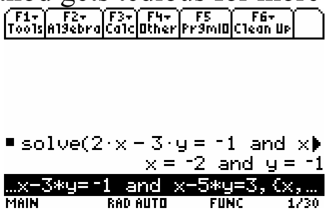


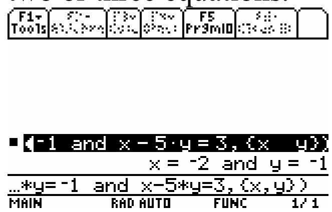




You can use matrices to solve this type of problem. Details are presented below in the matrix section.

Note that you can also use F2/Solve(to find solutions to simultaneous equations, although this method gets tedious for more than two or three equations.



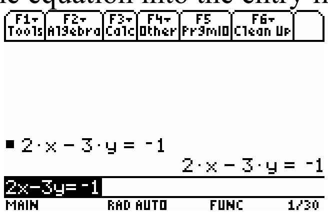


Note the brackets around *x*,*y*; i.e. $\{x,y\}$. You are solving for both variables, not just one. Recall that *and* is in the CATALOG.

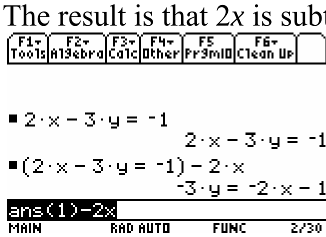
Doing Algebra

Solve $2x-3y=-1$ for *y* so the equation can be graphed.

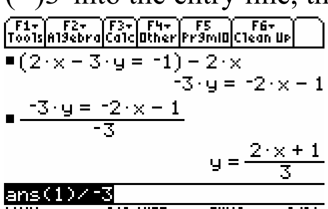
Enter the equation into the entry line, then press ENTER



Enter –2*x* into the entry line, then press ENTER. The TI supplies the ans(1), the previous result. The result is that 2*x* is subtracted from both sides of the equation.



Enter $\div(-)3$ into the entry line, then press ENTER. The TI supplies the ans(1).

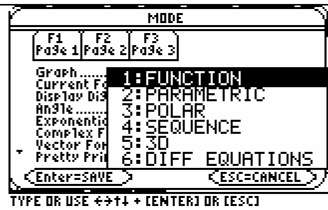


Using F2/Extract/Right, you can isolate the $\frac{2x+1}{3}$. Select, copy (F1/Copy), then paste (F1/Paste) into an empty *y*= slot.

Be sure to peruse the MATH/Number menu. There you can find items such as mod, least common multiplier, lcm(, greatest common divisor, gcd(, etc.

Graphing

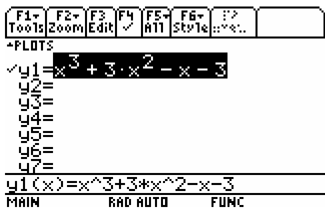
The TI-89 can generate six types of graphs. I will discuss Function, Parametric, Polar, and Diff Equations graphs.



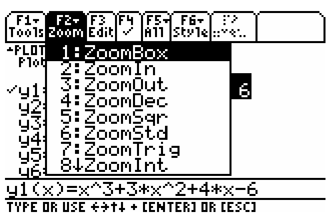
Function Graphs

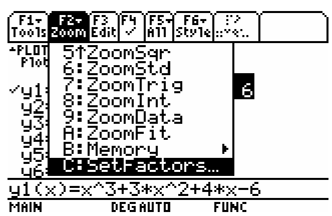
In order to create a graph of a function, you need to specify the function with line type as well as what window you will use to look at it. Typically, for math problems expected to hover near the origin, I create the function first, then let the calculator create the initial grid by selecting (F2/6: ZoomStd). This command sets the graph window to -10≤*x*≤10 and -10≤*y*≤10.

Graph $x^3 + 3x^2 - x - 3$. You can get to *y*= via the Apps menu as well as ♦ *F1*.



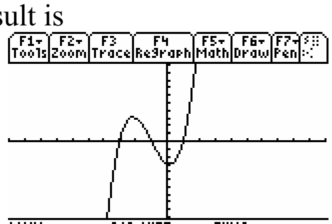
The *Zoom* (*F2*) menu gives the following screen, shown in two parts since there are more menu items available than will fit on one screen. Note the down arrow next to the 8th choice on the first screen. This arrow tells you to keep going, there are more options.



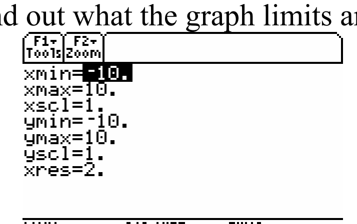


I choose *ZoomStd* as the equation should fit nicely within the -10≤x≤10 and -10≤y≤10 window.

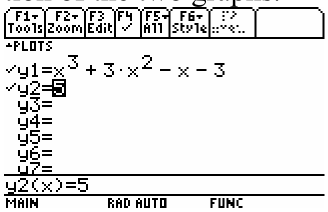
The result is

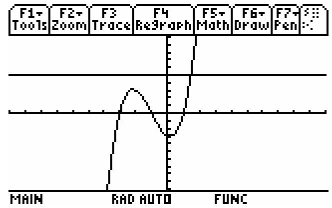


To find out what the graph limits are, press ♦ *F2*.

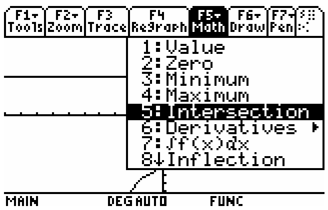


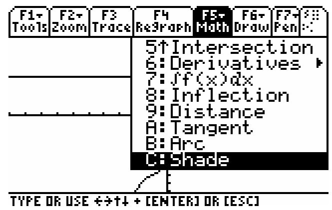
Suppose I now want to know for what value of *x* results in *y*=5. Set y2=5, then find the intersection of the two graphs.



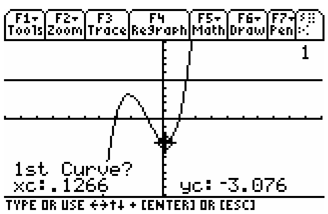


Choose *Intersection* from the current *Math* menu (*F5*). The choices for *F5* are

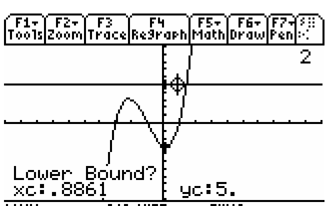


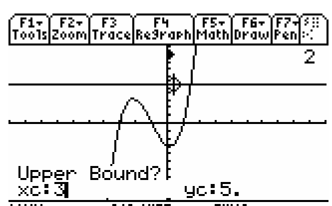


The calculator then asks which curves you want to intersect. The curve number is in the upper right-hand corner of the screen. To move among curves, use the up/down cursor arrows. Since we only have two curves, press ENTER twice.

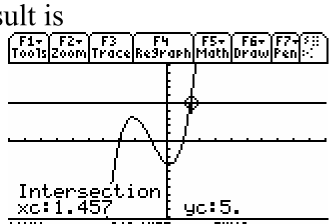


We have chosen Intersection, then which two curves to intersect. We are now presented with *Lower Bound?* The lower bound is any point to the left of the intersection and to the right of all other intersections to the left of the chosen one. You can move along a curve or type in a value for *x*. The upper bound establishes the maximum value of *x* that the calculator will use to find the intersection point. I moved along the curve for the lower bound and typed in *3* for the upper bound. Once I press ENTER, the calculator finds the intersection; hence, the upper bound selection does not show on the graph.

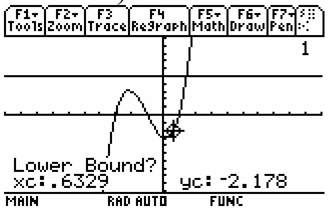


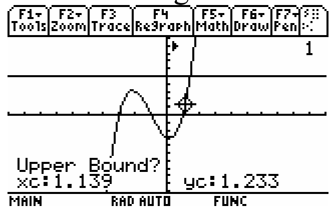


The result is

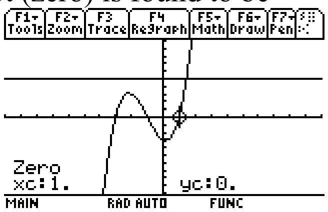


Now let's find the root of the function. Instead of *Intersection*, select *Zero* from the *Math* menu (F5/2:Zero). I choose to find the root on the right.

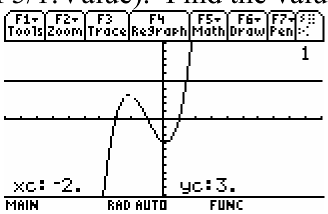




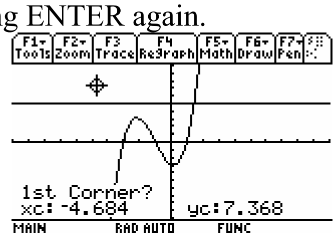
The root (zero) is found to be

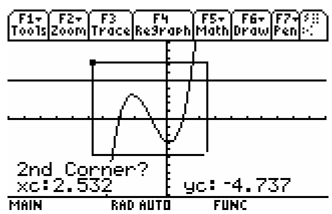


You can evaluate the function at a particular value of *x* by choosing *Value* from the *Math* menu (F5/1:Value). Find the value of the cubic when *x*=*-2*. Type -2, then press ENTER.

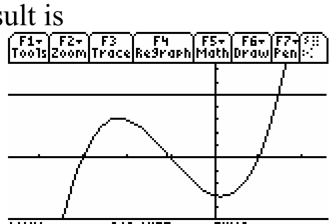


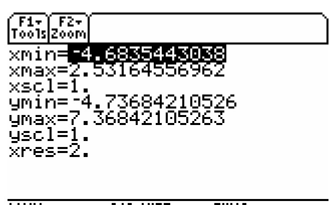
Let's now zoom in on a region. My favorite is *ZoomBox*. Select F2/1:ZoomBox. Move the cursor to the first corner, then press ENTER. Now create the box by moving the cursor, then pressing ENTER again.



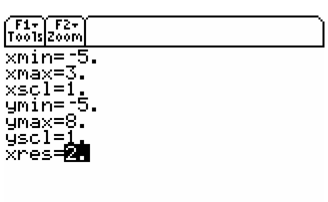


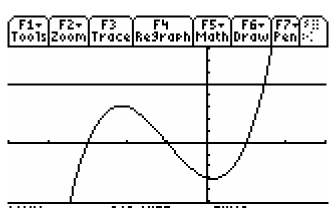
The result is





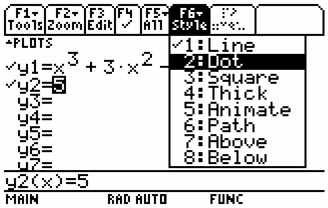
In this case we zoomed in on the intersection region. The new window coordinates are shown. If you prefer tidy graph limits, then edit the numbers in the Window screen (♦ F7).

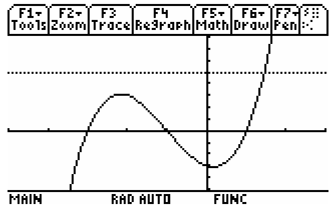




You should play with other types of zooming.

To select a line type, select the graph function in the *y*= section. *F6* provides a selection of line types.

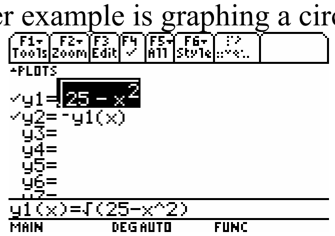


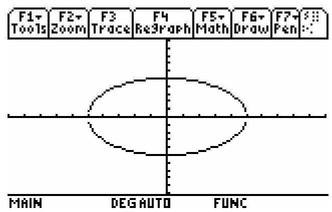


If you wish to suppress the graphing of a function but don't want to erase it, select that function, then *F4* in the *y*= screen. *F4* allows you to select (√). i.e. you want to graph it and deselect (no √) a function to not graph it.

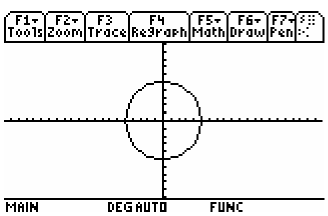
Note that you can create a complicated function on the entry line of the HOME screen, then COPY/PASTE it into one of the *y*'s.

Another example is graphing a circle.





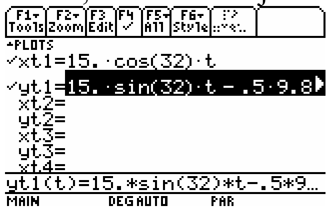
The circle looks like an ellipse due to the aspect ratio of the axes. Now use F2/5: ZoomSqr to correct for that effect.



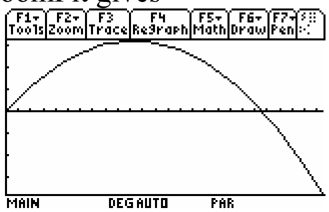
To plot data points, select Plot instead of *y*=. This choice is discussed in the Statistics - Regression section below.

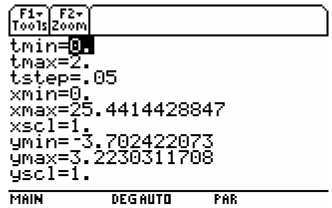
Parametric graphing

A classic case, besides Lissajous figures, is projectile motion:

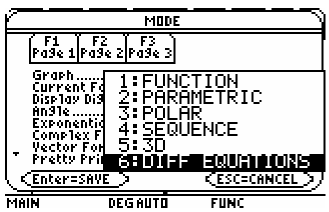


F2/A:ZoomFit gives

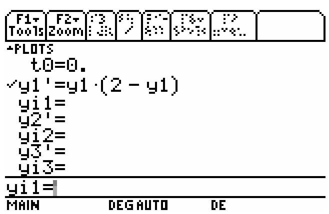




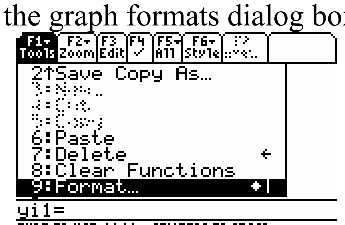
Differential equation graphing [pp 396-8 Electronic Guidebook]

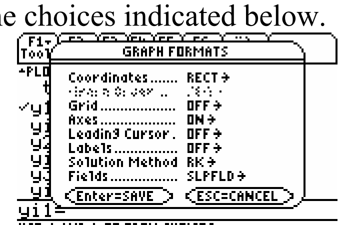


Select *y*= to enter the differential equation. Temporarily leave the initial condition, *yi1*, blank.

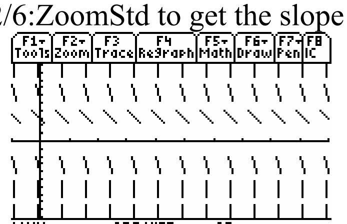


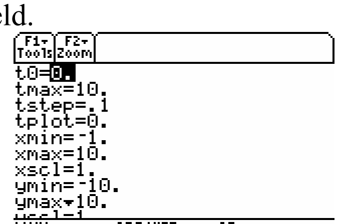
Select the graph formats dialog box. Make the choices indicated below.



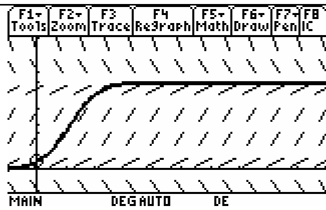


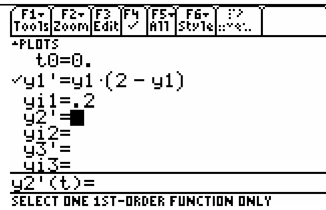
Use F2/6:ZoomStd to get the slope field.

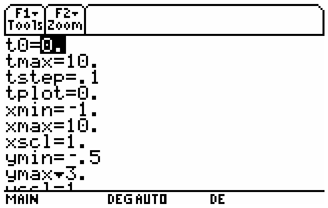




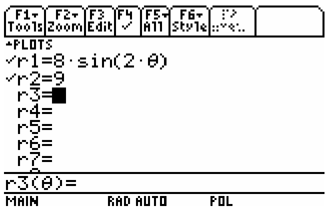
Now let *yi1*=0.2 to get the solution with that initial condition. You will then get a line that is the solution starting with that value at t0=0. I played a bit with the window to see the solution better.

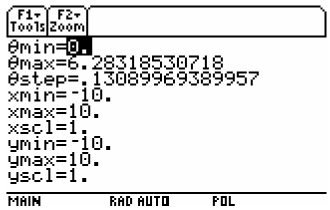






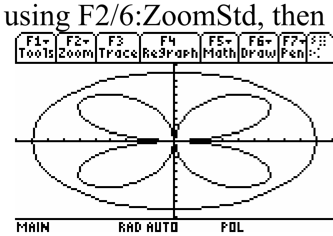
Polar graphing

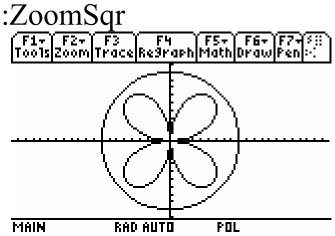




Note that if you type 2π for *θmax*=2π, the TI will evaluate the expression to its numerical equivalent. You can also type trig functions, etc., on the right hand side of the Window settings.

Graph using F2/6:ZoomStd, then F2/5:ZoomSqr





Trig

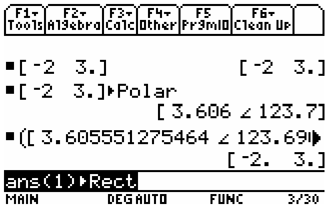
The three major trig buttons with inverses (*sin*, *cos*, *tan*) are on the main keyboard above the *Y, Z, T* buttons. The other three functions with inverses as well as the hyperbolic trig functions are available in the CATALOG. They can also be found in 2^{nd} 5 (the MATH button)/A:Trig or C:Hyperbolic.

Vectors and Matrices

The TI-89 has a full set of matrix algebra operations, such as augment, rref, diagonals, determinants, eigensystems, LU and QR decompositions, norms, row operations etc. This section discusses only those operations students typically use in many technical courses, not a full matrix algebra course like MTH 220. See the Guidebooks for more information.

Vectors

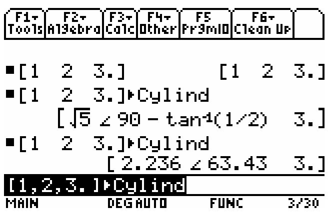
A vector is treated as a single row matrix. You can enter the vector data either using Apps/ Data/matrix editor or by entering from the entry line: [x, y, z]. You can also enter vectors in polar form: [2., 5.] or [3., ∠ 52]. **Conversions** between the two forms are straightforward (use CATALOG or MATH/Matrix/Vector ops). Note that I use a decimal in order to not get back the Exact form. Also note that I am in Degree mode (see the Status line).

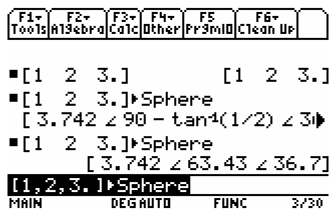


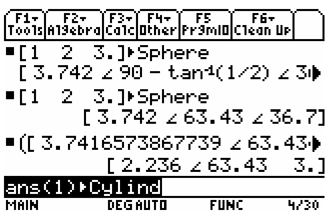
In addition, there are commands to extract the angle or radius, etc.

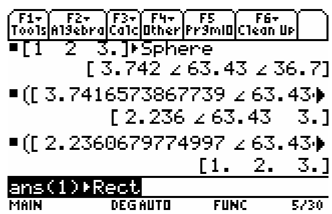
Vectors in 3D space are also straightforward to convert to cylindrical and spherical coordinates. The second conversion to cylindrical below was performed using ♦ ENTER. Pressing the ♦ in front of the ENTER guarantees an approximate solution, not exact. The exact result is shown in the first conversion. It is difficult to read and not too helpful unless you happen to know what the arctangent of ½ is.

To enter the vector you wish to convert, you can either use *Ans*(1), as above, or place the cursor at the end of the entry line before selecting the conversion operator from the catalog.

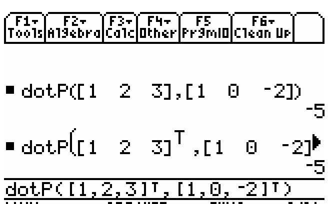


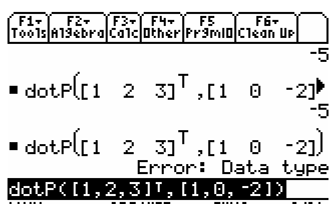


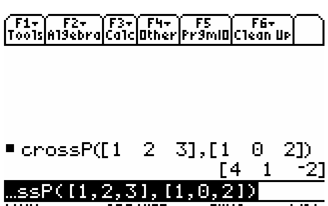


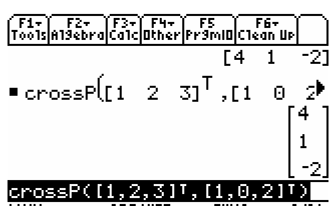


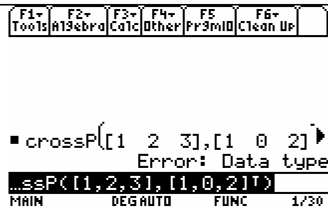
You can perform **dot products** and **cross products** and you can make a **unit vector**. These operations are found in the CATALOG as well as MATH/Matrix/Vector ops. The **transpose** operator (^{T}) is the first entry in MATH/Matrix. It is also the first entry under CATALOG/T. For both dot and cross products, both vectors must be either row or column vectors. You will get an error message if you mix types.

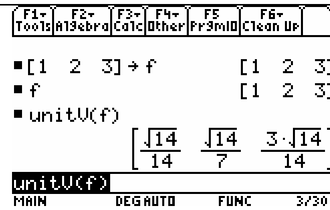




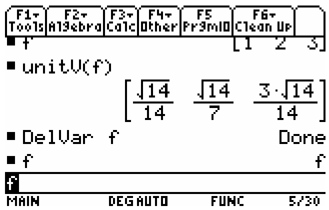




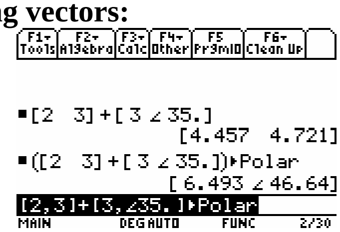




Note that when making the unit vector, I first stored the vector as variable name, *f*, for future use. Use single character names for temporary variables as you can clear them all using F6/1:Clear a-z from the home screen. You can delete individual letters using *Delvar* f (F4/4). I also let the format default to *Exact* instead of *Approximate* hence I got all those radical signs.

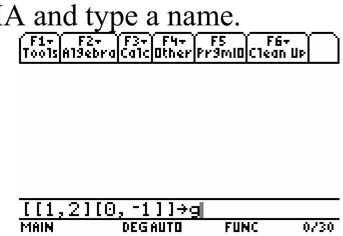


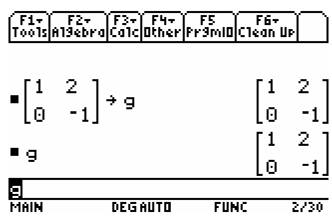
Adding vectors:



Matrices

You can enter the matrix data either using Apps/(Data/Matrix) editor or by entering from the entry line: [[a11, a12, a13], [a21, a22, a23], [a31, a32, a33]]. For easy editing later, you might want to store matrices. After entering it from the entry line, press STO►, then press ALPHA and type a name.

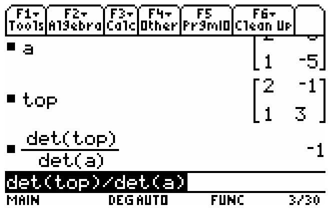




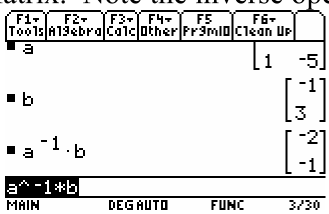
When you create a matrix using the Data/Matrix editor, you are initially given three options: List, Data and Matrix. From the Electronic Guidebook [1, pp501-4]:

> "A **list** is a series of items (numbers, expressions, or character strings) that may or may not be related. "
> "A **data** variable is essentially a collection of lists that may or may not be related." This data type looks like a spreadsheet to me.
> "A **matrix** is a rectangular array of elements." We will discuss this one.

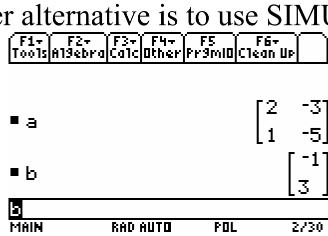
Once you have defined a matrix, you can perform all the MTH 220, Linear Algebra, operations on it. Let's use **Cramer's rule** to solve $\begin{array}{l} 2x-3y=-1 \\ x-5y=3 \end{array}$. Two matrices were defined using the Data/Matrix application (Apps/6). Once ***A*** was defined, I defined ***B*** by using F1/New to create another matrix while still in the matrix editor. The *det* operator is MATH/Matrix/det(as well as CATALOG/det(. Below I calculate the *y* value.

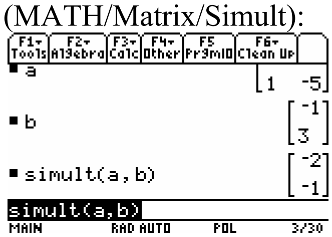


You can also solve the matrix equation $A\vec{x} = \vec{b}$ as follows. I defined ***A*** and ***b*** in Apps/6: Data/Matrix. Note the inverse operator is just raising the matrix ***A*** to the -1.

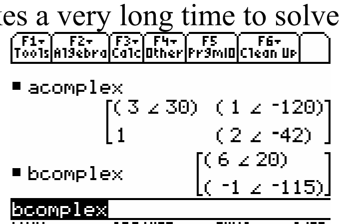


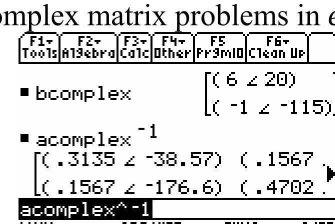
Another alternative is to use SIMULT (MATH/Matrix/Simult):

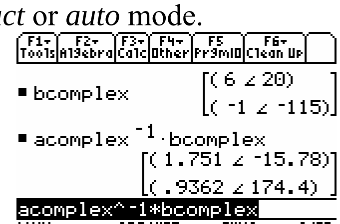




When the matrix elements are complex, you need to select your MODE settings carefully. According to [4, p 57], set Angle mode to DEGREE and ComplexFormat to POLAR. You can enter your elements in either polar ($r\angle\theta$) or rectangular a+jb form. For $j = \sqrt{-1}$, use *i* (2nd CATALOG) to get a+ib. I used ♦ ENTER to find the inverse as well as the solution. It takes a very long time to solve complex matrix problems in *exact* or *auto* mode.







Note: On occasion, you may want to write the complex equation $A\vec{x} = \vec{b}$ as purely real. You can make the problem purely real by using the following algorithm.

Given: $(A + Bi)(x + iy) = a + ib$ *A, B, a, b, x, y* are all real.

Expand into two equations: $Ax - By = a$ and $Bx + Ay = b$

Write in block matrix form (the matrix elements are themselves matrices):

$$\begin{pmatrix} A & -B \\ B & A \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}$$

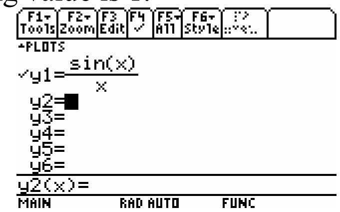
This real equation can be solved to find *x* and *y*: $\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} A & -B \\ B & A \end{pmatrix}^{-1} \begin{pmatrix} a \\ b \end{pmatrix}$.

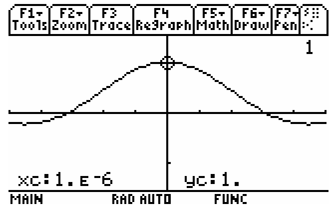
You could use the *Simultaneous Equation Solver* or *Simult* at this point as well.

Tables

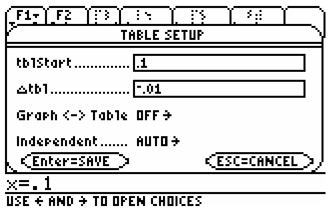
Tables are used to display a list of values for a given function.

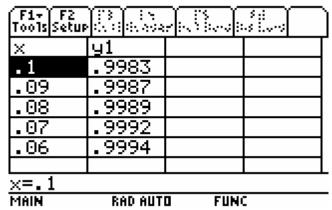
To set one up, first create the function. We will look at $y = \dfrac{\sin x}{x}$ as x goes to 0. The limiting value is 1.



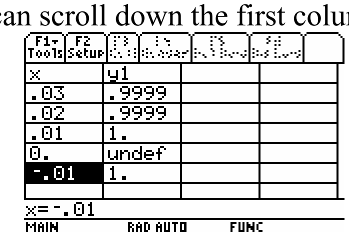


Now set up the initial table parameters using TblSet (♦ *F4*). ♦ *F5* displays the table generated.

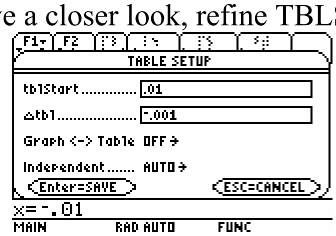


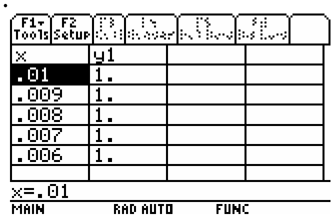


You can scroll down the first column until the desired number is found.



To have a closer look, refine TBLSET.





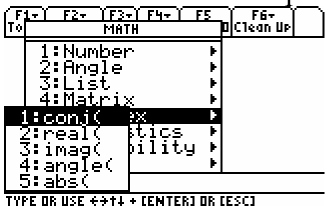
Complex Numbers

There are three forms in which to write complex numbers

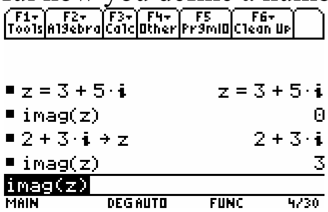
Rectangular	$a+ib$	radian or degree mode
Polar	$(r\angle\theta)$	radian or degree mode. *Note* - no comma - as in vectors
Exponential	$re^{i\theta}$	radian mode only

You may enter complex numbers in any format. Depending on your choice, Rectangular ($a+ib$), real (only uses complex notation if you enter complex numbers) or polar$(r\angle\theta)$, when you press ENTER, the number is converted to that format. ***Note***: complex operations always work in radian mode. See note 9 at the beginning of this document. "In Degree angle mode, complex identities such as e^(***i***q) = cos(q) + ***i*** sin(q) are not generally true because the values for cos and sin are converted to radians, while those for e^( ) are not. For example, e^(***i*** 45) = cos(45) + ***i*** sin(45) is treated internally as e^(***i***45) = cos(π/4) + ***i*** sin(π /4). Complex identities are always true in Radian angle mode." [p 912 Electronic Guidebook]

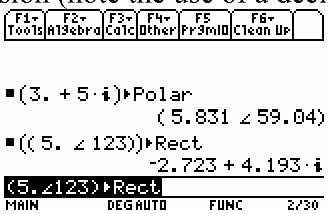
Operations in the MATH/Complex menu are



Be careful how you define a number as complex:



Conversion (note the use of a decimal point to suppress weird answers):



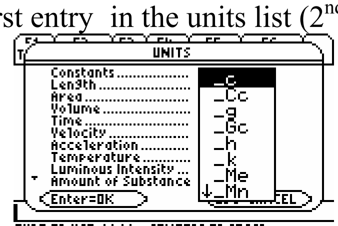
Assign keys 1-9 to applications

You can assign keys 1through 9 to run a program, as I mentioned above. For instance, suppose you have a program called RunMe. To assign the 3 key to this program you must rename RunMe to kbdprgm3(). You can then run it using ♦ 3. [pp 626-7 Electronic Guidebook]. Note that an example of *kbdprgm* is given on page 8 of this document.

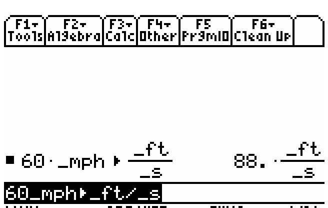
Unit creation and conversion

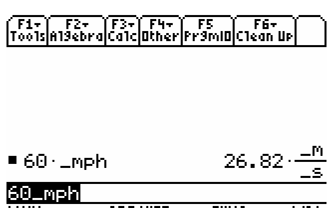
Unit conversion

The first entry in the units list (2nd 3) contains a bunch of built-in constants:

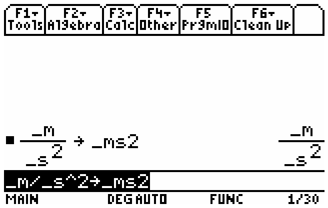


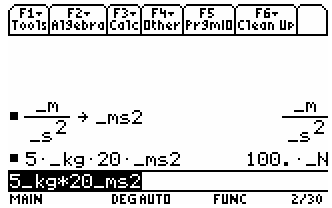
The TI-89 will convert units for you. The two unit operation characters, ► and _, are above the MODE key. The TI-89 converts to the default system (MODE/F3) if no final units are specified. Note that the unit ft/s is not defined in the *velocity* submenu so I had to use actual division of units.





Unit Creation [pp 52-3 Printed Guidebook]:
Create an acceleration unit called *_ms2*, then use it in *F=ma* to find force. Remember that the character → is what you get on the status line when you press the STO key. The result is given in *Newtons*, the SI unit of force.





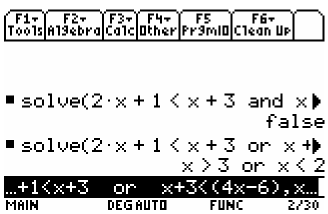
Bases

The options available under MATH/Base are

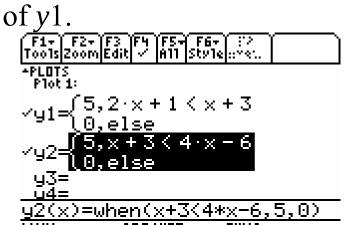
The default base is set in MODE/F2.

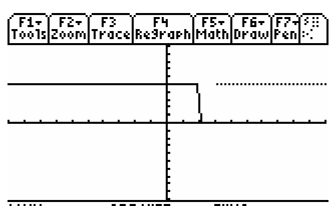
Inequalities

Solving inequalities
Solve $2x+1<x+3<4x-6$. The inequality must be split into two single inequalities. Try *and* first. If the answer is false, then switch to *or*.

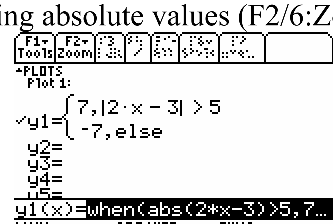


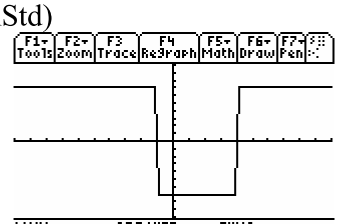
Graphing inequalities
Use the *When* function. The *when* function format is *When*(*condition, true, false*) just like the *if* function in Excel and in Word's Mail Merge. Note that I used *F6* to change the style of *y2* to dots. Using a line connects the last true point with the first false one, as shown in the graph of *y*1.





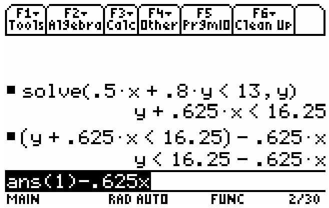
Graphing absolute values (F2/6:ZoomStd)

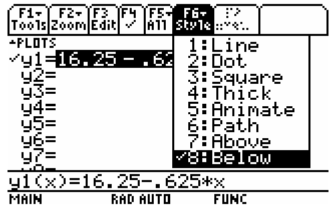


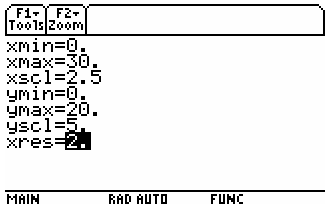


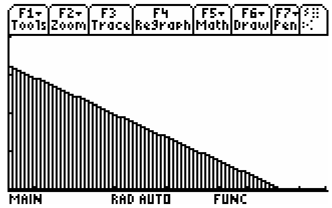
From Kelly's "Advanced Algebra with the TI-89", we wish to graph $0.5x + 0.8y \le 13$.

Solve for y algebraically, then graph. Since it turns out $y \le \frac{(13-0.5x)}{0.8}$, all values below the line $y = \frac{(13-0.5x)}{0.8}$ are valid solutions so should be shaded.





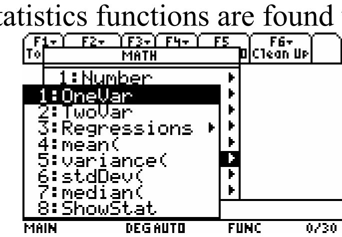




This type of plot is used in Linear Programming.

Statistics – Regression [pp 917-918 Electronic Guidebook]

The statistics functions are found under MATH/Statistics.



You find the regression functions under MATH/Statistics/Regressions. The simple version is

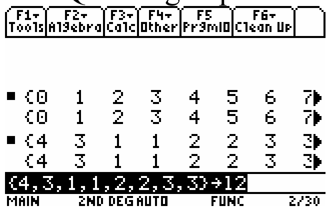
CubicReg xlist,ylist	$y= ax^3+ bx^2+ cx+ d$
ExpReg xlist,ylist	$y= ab^x$
LinReg xlist,ylist	$y= ax+ b$
LnReg xlist,ylist	$y= a+ b \ln(x)$
Logistic xlist,ylist	y=a/(1+b*e ^(c*x))+d i.e. $y = \frac{a}{1+be^{cx}} + d$
PowerReg xlist,ylist	$y=ax^b$
QuadReg xlist,ylist	$y= ax^2+ bx+ c$
QuartReg xlist,ylist	$y= ax^4+ bx^3+ cx^2+ dx+ e$
SinReg xlist,ylist	$y=a \sin(bx+c)+d$
MedMed xlist,ylist	calculates the median-median line

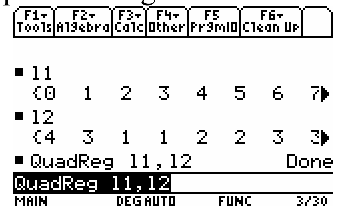
They can all accept more arguments. For example,

QuadReg list1**,** list2[**,** [list3] [**,** list4**,** list5]]

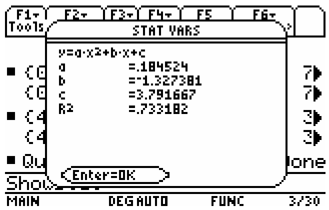
list1 represents **xlist**.
list2 represents **ylist**.
list3 represents **frequency**.
list4 represents **category codes**.
list5 represents **category include list**.

First I will use the example in the Electronic Guidebook [1], pp 832-3. Enter the two lists, *l1* and *l2* from the entry line using {}. Then use MATH/Statistics/Regressions/QuadReg or CATALOG/QuadReg to perform the quadratic regression.



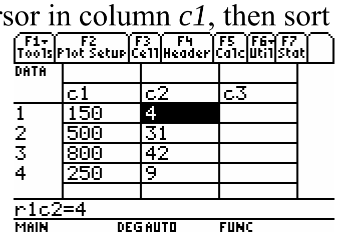


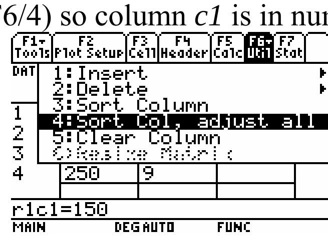
Finally, *ShowStat* (MATH/Statistics/6) to see the statistical information.



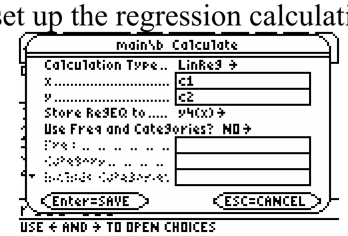
Now we want to enter data, then plot it including the regression line [pp 106-115 Electronic Guidebook].

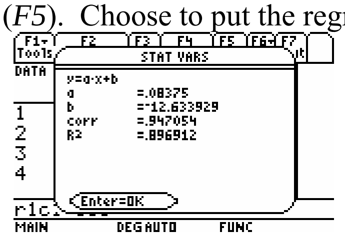
First enter the data into the Data/Matrix Editor – Apps/6. (It's Data). Enter your data, place the cursor in column *c1*, then sort it (F6/4) so column *c1* is in numerical order.



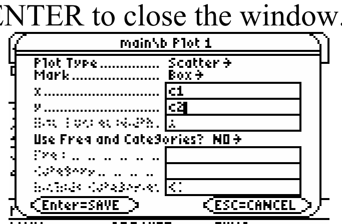


Now set up the regression calculation (*F5*). Choose to put the regression equation into *y*4(*x*).

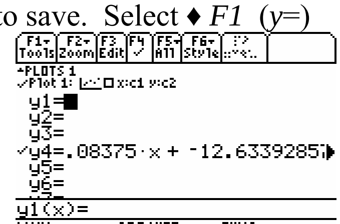




Press ENTER to close the window. Press *F2* for Plot Setup. Select *F1* (Define).

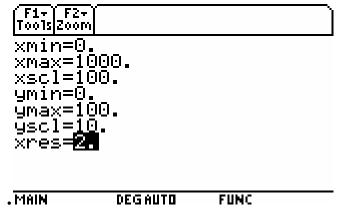


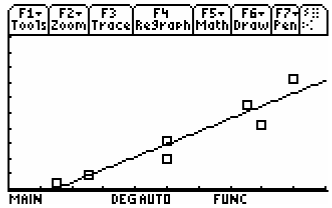
Enter to save. Select ♦ *F1* (*y*=)



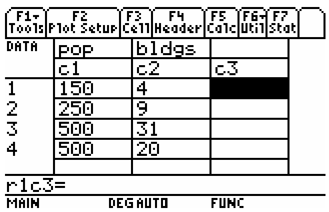
The data will be plotted in Plot 1 and the regression curve in *y*4(*x*).

Set up the plot window based on the data (or use F2/9:ZoomData), then GRAPH (♦ *F3*)

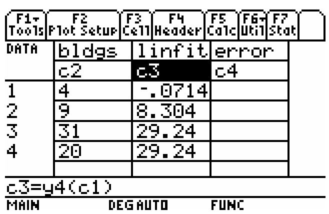


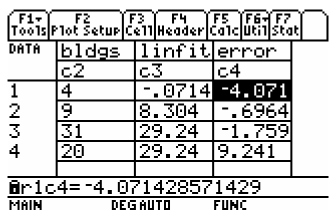


If you wish to name the data columns, return to the Data/Matrix editor. Label column *c1* by placing the cursor in the cell above the cell labeled *c1*, press ENTER. Type the title. Repeat for *c2*.

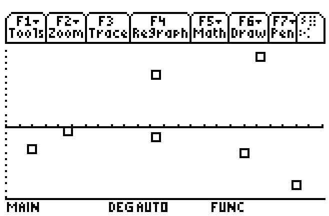


Now we calculate the error. Move the cursor to the cell named *c3*, then type the formula. Repeat for *c4*. Label both columns so you know what they are.





Now plot the residual error. Go to *y*= (♦ *F1),* and turn off (*F4*) *y*4 so the regression line is suppressed. Select F2/9:ZoomData



If you suspect another fit may do a better job, repeat all the above, writing the new data fit to a different *y* function, then compare the residuals of the two data fits using Plot1 and Plot2, for example, for the residual data. You just keep creating new columns in the data table.

TI-89 Books available for sale on the TI website. http://education.ti.com/us/activity/books/collegebooks.html	
	Electrical Engineering Applications with the TI-89 (This book is my Reference 4) by David R. Voltmer and Mark A. Yoder **Table of Contents** Activity 1: DC Circuit Analysis Activity 2: Transient Circuit Analysis: Symbolic Activity 3: Transient Circuit Analysis: Numeric Activity 4: Steady-State Circuit Analysis and Filter Design (147 KB)* Activity 5: Power Engineering Activity 6: Laplace Analysis: The s-domain Activity 7: The Convolution Activity 8: Fourier Series Activity 9: Vectors Activity 10: Vector Calculus Activity 11: Electromagnetics Activity 12: Transmission Lines Activity 13: Antennas Activity 14: Manipulating Lab Data: The Diode Activity 15: Financial Calculations Index
	Advanced Placement Calculus with the TI-89 by Ray Barton and John Diehl **Table of Contents** Activity 1: Functions, Graphs, and Limits Activity 2: Differentiation Activity 3: Applications of the Derivative Activity 4: Integration (65 KB)* Activity 5: Riemann Sums and the Fundamental Theorem of Calculus Activity 6: Applications of Integrals (122 KB)* Activity 7: Differential Equations and Slope Fields Activity 8: Parametric, Vector, Polar, and 3D Functions Activity 9: Infinite Sequences and Series Appendix A: TI-89 Keystrokes and Menus Appendix B: Common Calculus Operations Appendix C: Creating Scripts Appendix D: Solutions to the Exercises Index

Major-Specific Applications Available for free on the TI website.	
	MEPro – see attached Table of Contents
	EEPro – see attached Table of Contents
	EE200 – see attached Table of Contents