TI-85 QUICK REFERENCE GUIDE

By Gary A. Harris
Department of Mathematics
Texas Tech University
Lubbock, Texas 79409
g.harris@ttu.edu

GRAPH package (rectangular coordinates)

y(x)=	*Place to enter and edit functions to be graphed*
x	*Independent variable*
y	*Dependent variable (needs integer to indicate which previously defined function to use)*
INSf	*Inserts a function into the list if space permits*
DELf	*Deletes designated function from list*
SELCT	*On/Off switch to determine which functions in list are to be graphed*
ALL+	*Turns all functions on*
ALL-	*Turns all function off*

RANGE	*Used to manually set viewing window*
xMin=	*Sets minimum x*
xMax=	*Sets maximum x*
xScl=	*Sets spacing between hash marks on x axis*
yMin=	*Sets minimum y*
yMax=	*Sets maximum y*
yScl=	*Sets spacing between hash marks on y axis*

Observations:

1. There are 127 pixels horizontally across the screen. So the step size for sampling x is

$$h = (xMax - xMin) / 126 .$$

2. Arithmetic operations can be used in defining the viewing window. For example, if you wish to sample evey 5 integers starting with 0, simply set

$$xMin = 0 \text{ and } xMax = 5(126).$$

Note that xScl=1 is not very good for this window, xScl=50 might be better. On the other hand, if you want the sample step size to be .01 with 0 at the center set

$$xMin=-126/200 \text{ and } xMax=126/200.$$

3. The fact that 126 is even tells us that the center of the interval [xMin , xMax] is always a sample point because

$$xMin + 63\ h = (xMax + xMin) / 2.$$

ZOOM *Used to automatically change viewing window*

BOX *Creates a view box which becomes viewing window (Position cursor and ENTER to create one corner, then position cursor and ENTER to create opposite corner)*

ZIN *Zooms in centered on cursor position*

ZOUT *Zooms out centered on cursor position*

ZSTD *Sets default viewing window (xMin = yMin = -10, xMax = yMax = 10, xScl = yScl = 1)*

ZPREV *Returns to previous viewing window*

ZFIT *Choses "best window" to fit the graph*

ZTRIG *Choses "best window" for viewing trig graphs*

ZDECM *Choses xMin = - 6.3 & xMax = 6.3 so 0 is the center and h = .1)*

ZRCL *Recalls a stored window*

ZFACT *Used to set the "Zoom factors" (default is 4 for x and y, i.e. ZIN divides the x and y distances by 4)*

ZOOMX *Multiplies x-distance by x-factor*

ZOOMY *Multiplies y-distance by y-factor*

ZINT *Sets xMin = -63 & xMax = 63 (so h = 1)*

ZSTO *Stores the current window*

TRACE *Begins with cursor at the center of the graph of y1*
Rt arrow traces to right in increments of h
Lt arrow traces to left in increments of h
Dn arrow jumps to the graph of y2 with same x-coord .
Etc. (x and y coordinates of cursor appear on screen)

GRAPH *Graphs all functions that are turned on (in current window)*

MATH *Contains most of the mathematical concepts of Cal I*

LOWER *Sets a lower bound on x*

UPPER *Sets an upper bound on x*

ROOT *Searches for a root within the bounds on x*

dy/dx *Computes the derivative at the cursor position on the graph (TRACE is automatically activated)*

f(x) *Computes the integral between lower limit (1st ENTER) and upper limit (2nd ENTER) (TRACE is automatically activated)*

FMIN *Computes the min of f within the bounds on x*

FMAX *Computes the max of f within the bounds on x*

INFLC *Seeks an inflection point of f within the bounds on x*

YICPT *Seeks the y intercept*

ISECT *Seeks an intersection point between two graphs within the bounds on x (TRACE is automatically*

activated and used to estimate the point and determine the two graphs)

DIST *Computes the straight line distance between two points on a graph (TRACE is automatically activated)*

ARC *Computes the arc length between two points on a graph (TRACE is automatically activated)*

TANLN *Draws the tangent line to a point on a graph and displays its slope (TRACE is automatically activated)*

DRAW *Used for drawing pictures but appears to have little computing ability*

Shade *Graphs and shades the area between two graphs (Shade(y1,y2,a,b) graphs y1, y2, and shades in the region above y1, below y2, from x = a to x = b)*

LINE *Draws the line segment between two points (1st cursor position ENTER, and second cursor position ENTER)*

VERT *Draws vertical line through cursor position (with as many positions as you wish $\leq$ 126)*

CIRCL *Draws a circle with center the lst cursor position ENTER and radius determined by the second*

DrawF *Draws the graph of F (DrawF y1)*

PEN *Allows the cursor to be used as a pen*

PTOFF *Turns the designated pixel off*

PTON *Turns the designated pixel on*

PTCHG *Turns on pixel off and off pixel on*

CLDRW *Clears all the drawn stuff from the screen (returns the graphed stuff)*

TanLn *Draws the tangent line to the specified function at the specified value of x (TanLn(y1, x) draws the tangent line to y1 at (x,y1(x))*

DrInv *Draws the inverted graph of the specified function (DrInv y1 draws the inverted graph of y1)*

FORMT *Lets you change various graphing format options*

RectGC PolarGC *Displays coordinates in rectangular of polar form*

CoordOn CoordOff *Displays coordinates or does not display coords*

DrawLine DrawDot *Connects points with lines or does not*

SeqG SimulG *Graphs functions in order or simultaneously*

GridOff GridOn *Does not display coord grid or does display it*

Osazone AxesOff *Displays coord axes or does not display them*

LabelOff LabelOn *Does not label x & y or does label x & y*

STGDB	*Stores current graphs up to 5 (data and all)*
RCGDB	*Recalls any of the stored graphs (data and all)*
EVAL	*Evaluates y for given x (can jump between graphs)*
STPIC	*Stores current picture up to 5*
RCPIC	*Recalls any of the stored pictures*