

# **Graphing Calculators Across the Calculus Curriculum**

*AMS 2000 Spring Eastern Sectional Meeting*

*Lowell, MA*

*April 1-2, 2000*

Marvin Stick

Department of Mathematical Sciences

University of Massachusetts Lowell

**University of Massachusetts Lowell**  
**Department of Mathematical Sciences**  
1 University Avenue  
Lowell, MA 01854

(978)934-2441  
FAX (978)934-3053  
Email: Marvin\_Stick@uml.edu

Special Session on Teaching Mathematics in the New Millenium, III

**Abstract:** *Technology approaches will be presented to facilitate the teaching and learning of calculus related material from precalculus to the multivariate level. Distinct approaches and calculator techniques used for applied mathematics courses in Management and those used for Science and Engineering students will be presented. TI calculator code listings are supplied.*

The procedures and programs represent tools I have used while teaching different levels of calculus related material. Many of the programs and procedures were developed from scratch, while others were adapted from existing materials. The code is for the TI-83 Plus group of graphing calculators.

The sampling of calculator tools presented focus on issues that I have addressed at various levels. In addition to the tools presented, I have found that most students gain a better understanding of function behavior and the underlying mathematical concepts when manipulating window domain and range settings as opposed to using automated window setting features. I have found this approach most helpful for all types of limit problems.

The focus of the following material is with the use of 2D graphing capability.

# Contents

<b>Algebra</b>	
1. Inequalities .....	4
2. Rational Equations .....	4
<b>Precalculus</b>	
1. Radical Equations .....	5
2. System of Equations .....	5
3. Conics .....	6
Parabola .....	6
Ellipse .....	8
Hyperbola.....	10
<b>Differential Calculus</b>	
1. Local Max/Min .....	12
2. Point of Diminishing Return .....	12
3. Inflection Point .....	13
4. Mean Value Theorem .....	13
5. Price Elasticity of Demand .....	14
<b>Integral Calculus</b>	
1. Midpoint Rule .....	14
2. Differential Equations .....	16
3. Arc Length .....	17
Arc Length - Parametric Equations.....	17
4. Series.....	18
<b>Multivariable Calculus</b>	
1. Dot Product .....	18
2. Cross Product.....	19
3. Slices .....	20
4. Partial Derivatives.....	21
5. Iterated Integral .....	22
<b>Appendix A</b>	
Polynomial Approximations .....	23
1. $\sin(x)$ .....	23
2. $e^x$ .....	24
3. $\ln(x)$ .....	25
4. TI-89 Implementation .....	25

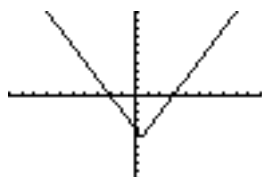
## Algebra

- Inequalities

- $|ax + b| \leq c$   $y_1 = abs(ax + b) - c$  and examine where  $y_1 \leq 0$

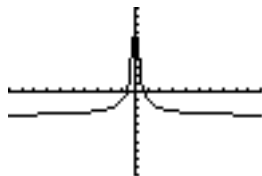
- $|ax + b| \geq c$   $y_1 = abs(ax + b) - c$  and examine where  $y_1 \geq 0$

ex:  $|2x - 1| \geq 5$ , define  $y_1 = abs(2x - 1) - 5$  and look for regions where  $y_1 \geq 0$



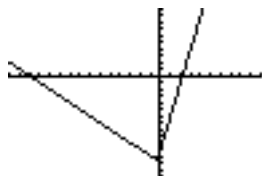
The following more analytically challenging examples may arise in follow-up courses when reviewing precalculus topics. The TI-83 approach is the same as that used above.

ex:  $\left|\frac{2}{x}\right| \leq 3$ , define  $y_1 = abs(2/x) - 3$  and look for regions where  $y_1 \leq 0$  to find that  $-\infty < x \leq -\frac{2}{3}$   
or  $\frac{2}{3} \leq x < \infty$



Analytically, consider separate cases for  $x > 0$  and  $x < 0$ .

ex:  $|3x + 1| < 2|x - 6|$ , define  $y_1 = abs(3x + 1) - 2abs(x - 6)$  and examine the graph



with a window set as Xmin=-15, Xmax=10, Ymin=-15, Ymax=10 to find that  $y_1 < 0$

when  $-13 < x < \frac{11}{5}$ .

Analytically, square both sides to get  $(3x + 1)^2 < 4(x - 6)^2$ . Collect terms and factor to get  $(5x - 11)(x + 13) < 0$ .

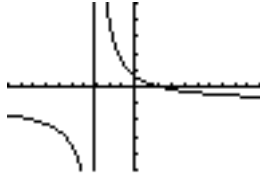
- Rational Equations

ex:  $\frac{5}{x - 3} - \frac{30}{x^2 - 9} = 1$

- Graphically:  $y_1 = 5/(x - 3) - 30/(x^2 - 9) - 1$

a) Look for zeroes of  $y_1$  (zoom 6)

b) Redefine window to ymin= -5 and ymax=5



- Analytically:  $\frac{5}{x-3} - \frac{30}{x^2-9} = 1$  LCD is  $(x+3)(x-3)$

$$(x+3)(x-3)\frac{5}{x-3} - (x+3)(x-3)\frac{30}{x^2-9} = 1(x+3)(x-3)$$

$$5x + 15 - 30 = x^2 - 9$$

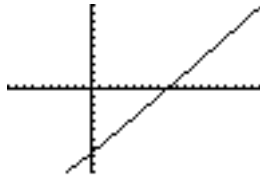
then  $x^2 - 5x + 6 = 0$  and  $x=2$  or  $x=3$ , but  $x=3$  is not a solution

## Precalculus

- Radical Equations

ex: Solve  $x = \sqrt{x+7} + 5$

- Graphically:  $y_1 = x - \sqrt{x+7} - 5$  and look for zeroes with  $X_{\min}=-10$ ,  $X_{\max}=20$ ,  $Y_{\min}=-10$ ,  $Y_{\max}=10$ .



- Analytically:  $x - 5 = \sqrt{x+7}$ , then square to get

$$(x-5)^2 = x+7 \quad \text{and} \quad x^2 - 10x + 25 = x+7 \quad \text{or} \quad x^2 - 11x + 18 = 0$$

then  $x=2$  or  $x=9$ , but  $x=2$  is not a solution

- Systems of Equations

$$ax + by = c$$

- $dx + ey = f$

- $y_1 = y_2$ : 2nd CALC intersect for a 2x2 system, for an nxn system introduce alternate techniques.

$$\begin{cases} AX = B \\ X = A^{-1}B \end{cases}$$

$$3x + 2y = 7$$

ex:

$$5x - y = 3$$

a) MATRX, -> to EDIT, 1 to edit matrix A

b) 2 enter (row), 2 enter (column), 3 enter, 2 enter, 5 enter, -1 enter, then 2nd QUIT

c) MATRX, -> to EDIT, 2 to edit matrix B

d) 2 enter (row), 1 enter (column) 7 enter, 3 enter, then 2nd QUIT

e) to compute  $X = A^{-1}B$ , MATRX 1 (i.e. matrix A),  $x^{-1}$  (inverse of A),

\* (multiplication symbol is optional) MATRX 2 (i.e. matrix B), enter

output is  $\begin{bmatrix} 1 \\ 2 \end{bmatrix}$

f) MATRX, 1,  $x^{-1}$  enter, yields  $A^{-1}$

ex: show that  $AA^{-1} = I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ : MATRX, NAMES, 1 (i.e. matrix A)

then \* (optional), MATRX, NAMES, 1 (i.e. matrix A),  $x^{-1}$  (inverse of A), enter

g) to output  $A^{-1}$  in a "pretty print" form, print the inverse matrix in decimal form as in step f), then Math, option 1, to get

```
Ans+Frac
[[1/13 2/13 ]
 [5/13 -3/13]]
■
```

- Conics

The following programs dealing with nonrotated conics can be used throughout the calculus curriculum. Normally the material is first introduced at a precalculus level. The examples used are also presented in the text files *conicel.txt*, *conichyp.txt*, *conicpx.txt*, *conicpy.txt* at the web site [www.ti.com](http://www.ti.com).

This program examines a parabola of the form  $y = ax^2 + bx + c$ . Input requirements are the coefficients. Output consists of the vertex, focus and directrix.

Program:PARABY

:ClrHome

:Disp "Y=AX^2+BX+C"

:Disp "A="

:Input A

:Disp "B="

:Input B

:Disp "C="

:Input C

:B/A->D

:1/A->F

:-D/2->H

:C-(B^2)/(4A)->K

:F/4->P

:Disp "VERTEX AT X="

:Disp H

:Disp "Y="

:Disp K

:Pause

:Disp "FOCUS AT X="

:Disp H

:Disp "Y="

:Disp K+P

:Disp "DIRECTRIX IS Y="

:Disp K-P

X coordinate of the vertex

Y coordinate of the vertex

Distance from focus to vertex

ex:  $y = -\frac{1}{2}x^2 - x + \frac{1}{2}$

output is: vertex at  $x = -1$   $y = 1$ , focus at  $x = -1$   $y = .5$ , directrix is  $y = 1.5$

This program examines a parabola of the form  $x = ay^2 + by + c$ . Input requirements are the coefficients. Output consists of the vertex, focus and directrix.

Program:PARABX

```

:ClrHome
:Disp "X=AY^2+BY+C"
:Disp "A="
:Input A
:Disp "B="
:Input B
:Disp "C="
:Input C
:B/A->D
:1/A->F
:-D/2->K
:C-(B^2)/(4A)->H
:F/4->P
:Disp "VERTEX AT X="
:Disp H
:Disp "Y="
:Disp K
:Pause
:Disp "FOCUS AT X="
:Disp H+P
:Disp "Y="
:Disp K
:Disp "DIRECTRIX IS X="
:Disp H-P

```

Y coordinate of the vertex  
X coordinate of the vertex  
Distance from the focus to vertex

ex:  $x = \frac{1}{2}y^2 + \frac{1}{2}y + \frac{41}{8}$

output is: vertex at  $x=5.5$   $y=-.5$ , focus at  $x=5.5$   $y=-.5$ , directrix is  $x=4.5$

The following program PARABOLA combines the features of the two previous versions. An advantage to looking at PARABY or PARABX separately is their relative simplicity.

Program:PARABOLA

```

:ClrHome
:Disp "IF Y=AX^2+BX+C"
:Disp "INPUT CODE=1"
:Disp "IF X=AY^2+BY+C"
:Disp "INPUT CODE=2"
:Disp "CODE="
:Input M
:Disp "A="
:Input A
:Disp "B="
:Input B
:Disp "C="
:Input C
:B/A->D
:1/A->F
:-D/2->H
:C-(B^2)/(4A)->K
:F/4->P
:If M=1

```

```

:Then
:1->X
:0->Y
:Else
:0->X
:1->Y
:End
:XH+YK->S
:XK+YH->T
:Disp "VERTEX AT X=",S
:Disp "Y=",T
:Pause
:ClrHome
:Disp "FOCUS AT X=",S+YP
:Disp "Y=",T+XP
:If M=1
:Then
:Disp "DIRECTRIXIS Y=",K-P
:Else
:Disp "DIRECTRIX IS X=",K-P
:End

```

The following program examines an ellipse in the form  $Ax^2 + Cy^2 + Dx + Ey + F = 0$ . Input consists of the coefficients. Output consists of the center, foci, vertices, end points of the minor axis and eccentricity. The program will also output a, b and c values for the standard form of the equation of an

ellipse  $\frac{(x-h)^2}{a^2} + \frac{(y-k)^2}{b^2} = 1$  with center at  $(h,k)$  if the major axis is horizontal and

$\frac{(x-h)^2}{b^2} + \frac{(y-k)^2}{a^2} = 1$  with center at  $(h,k)$  if the major axis is vertical. In both cases  $c^2 = a^2 - b^2$

and the major axis has length  $2a$ .

Program: ELLIPSE

```

:ClrHome
:Disp "AX^2+CY^2+DX"
:Disp "+EY+F=0"
:Disp "A="
:Input A
:Disp "C="
:Input C
:Disp "D="
:Input D
:Disp "E="
:Input E
:Disp "F="
:Input F
:-D/(2A)->H
:-E/(2C)->K
:(A(H^2)+C(K^2)-F)->M
:M/A->N
:M/C->O
:1->X
:0->Y
:If N<0
:Then
:0->X

```

Procedure to complete the square

"

Compute  $a^2$  and  $b^2$  for the standard form of the equation of an ellipse

Major axis is vertical  
Algorithm to adjust for output when



```

:1->Y
:√(O - N) ->C
:Else
:√(N - O) ->C
:End
:√N ->A
:√O ->B
:ClrHome
:Disp "CENTER AT"
:Disp "X=",H
:Disp "Y=",K
:Pause
:ClrHome
:Disp "A=",XA+YB
:Disp "B=",XB+YA
:Disp "C=",C
:Pause
:ClrHome
:Disp "FOCI AT"
:Disp "X=",H+XC
:Disp "Y=",K+YC
:Pause
:ClrHome
:Disp "AND X=",H-XC
:Disp "Y=",K-YC
:Pause
:ClrHome
:Disp "VERTICES AT"
:Disp "X=",H+XA
:Disp "Y=",K+YB
:Pause
:ClrHome
:Disp "AND X=",H-XA
:Disp "Y=",K-YB
:Pause
:ClrHome
:Disp "END POINTS FOR"
:Disp "MINOR AXIS"
:Disp "X=",H+YA
:Disp "Y=",K+XB
:Pause
:Clrhome
:Disp "AND X=",H-YA
:Disp "Y=",K-XB
:Pause
:ClrHome
:If A=0
:Then
:Disp "DEGENERATE"
:Disp "ELLIPSE"
:Else
:Disp "ECCENTRICITY=",C/(XA+YB)
:End

```

the major axis is vertical

Major axis is horizontal

Output for the standard form of the equation of an ellipse. 'a' is associated with the major axis, 'b' with the minor axis, and  $c^2 = a^2 - b^2$

ex:  $4x^2 + y^2 - 8x + 4y - 8 = 0$

output is: center at x=1 y= -2, a=4 b=2 c=3.464, foci at x=1 y=1.464 and x=1 y= -5.464, vertices at x=1 y=2 and x=1 y= -6, end points for minor axis x=3 y= -2 and x= -1 y= -2, eccentricity= .866

The following program examines a hyperbola in the form  $Ax^2 + Cy^2 + Dx + Ey + F = 0$ . Input consists of the coefficients. Output consists of the center, foci, vertices, slopes and intercepts of the asymptotes, end points of the conjugate axis and eccentricity. The program will also output a, b and c values for the standard form of the equation of a hyperbola  $\frac{(x-h)^2}{a^2} - \frac{(y-k)^2}{b^2} = 1$  with center at  $(h,k)$  if the transverse axis is horizontal and  $\frac{(y-k)^2}{a^2} - \frac{(x-h)^2}{b^2} = 1$  with center at  $(h,k)$  if the transverse axis is vertical. In both cases  $c^2 = a^2 + b^2$ . An exception case in the output will flag a degenerate hyperbola when  $a = b = c = 0$ .

Program: HYPERB

```

:ClrHome
:Disp "AX^2+CY^2+DX"
:Disp "+EY+F=0"
:Disp "A="
:Input A
:Disp "C="
:Input C
:Disp "D="
:Input D
:Disp "E="
:Input E
:Disp "F="
:Input F
:-D/(2A)->H
:-E/(2C)->K
:(A(H^2)+C(K^2)-F)->M
:M/A->N
:M/C->O
:1->X
:0->Y
:If N<0
:Then
:-N->N
:0->X
:1->Y
:Else
:-O->O
:End
:√N->A
:√O->B
:√(N+O)->C
:ClrHome
:Disp "CENTER AT"
:Disp "X=",H
:Disp "Y=",K
:Pause
:ClrHome
:Disp "A=",XA+YB
:Disp "B=",XB+YA

```

Procedure to complete the square

"

Compute  $a^2$  and  $b^2$  for the standard form of the equation of a hyperbola

Transverse axis is vertical

Algorithm to adjust for output when the transverse axis is vertical

Transverse axis is horizontal

Output for the standard form of the equation of a hyperbola. 'a' is associated with the transverse axis,

```

:Disp "C=",C
:Pause
:ClrHome
:Disp "FOCI AT"
:Disp "X=",H+XC
:Disp "Y=",K+YC
:Pause
:ClrHome
:Disp "AND X=",H-XC
:Disp "Y=",K-YC
:Pause
:ClrHome
:Disp "VERTICES AT"
:Disp "X=",H+XA
:Disp "Y=",K+YB
:Pause
:ClrHome
:Disp "AND X=",H-XA
:Disp "Y=",K-YB
:Pause
:ClrHome
:If A=0
:Then
:Disp "DEGENERATE"
:Disp "HYPERBOLA"
:Else
:Disp "ASYMPTOTES ARE"
:B/A->R
:Disp "Y1 INTERCEPT=",K-HR
:Disp "Y1 SLOPE",R
:Pause
:ClrHome
:Disp "Y2 INTERCEPT=",K+HR
:Disp "Y2 SLOPE=",-R
:End
:Pause
:ClrHome
:Disp "END POINTS OF"
:Disp "CONJUGATE AXIS"
:Disp "X=",H-YA
:Disp "Y=",K-XB
:Pause
:Clrhome
:Disp "AND X=",H+YA
:Disp "Y=",K+XB
:Pause
:ClrHome
:If A ≠ 0
:Disp "ECCENTRICITY=",C/(XA+YB)

```

'b' with the conjugate axis, and  $c^2 = a^2 + b^2$

Exception case to flag a degenerate hyperbola

ex:  $4x^2 - 3y^2 + 8x + 16 = 0$

output is: center at  $x = -1$   $y = 0$ ,  $a = 2$   $b = 1.732$   $c = 2.646$ , foci at  $x = -1$   $y = 2.646$  and  $x = -1$   $y = -2.646$ , vertices at  $x = -1$   $y = 2$  and  $x = -1$   $y = -2$ , asymptotes are  $y_1 = 1.155x + 1.155$

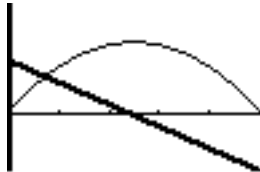
$y_2 = -1.155x - 1.155$ , end points of conjugate axis  $x = -2.732$   $y = 0$  and  $x = .732$   $y = 0$ , eccentricity = 1.323

## Differential Calculus

- Local Max/Min:  $s = \frac{1}{2}gt^2 + v_0t + s_0$

- $y_1 = f(x)$  and  $y_2 = nDeriv(y_1, x, x)$

ex:  $s = -16t^2 + 80t + 5$       Xmin=0, Xmax=5, Ymin= -80, Ymax=160  
Find zero for  $y_2$ , i.e. Intermediate Value Theorem



- Point of Diminishing Return:  $f(x) = 4\left(1 + \frac{x-4}{\sqrt{x^2 - 8x + 20}}\right)$

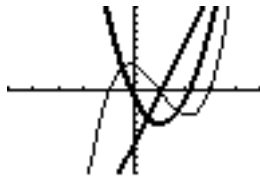
Xmin=0, Xmax=12, Ymin= -4, Ymax=10, Xres=2

- $y_1 = f(x)$     $y_2 = nDeriv(y_1, x, x)$     $y_3 = nDeriv(y_2, x, x)$



- Inflection Point:  $f(x) = x^3 - 3x^2 - x + 3$   
Xmin= -5, Xmax=5, Ymin= -10, Ymax=10, Xres=2

- $y_1 = f(x)$     $y_2 = nDeriv(y_1, x, x)$     $y_3 = nDeriv(y_2, x, x)$



- Mean Value Theorem

The program MVT will aid in showing the existence of points  $c$  where the slope of the secant line and tangent are approximately equal. However, the program will not work properly for cases such as those when  $f(x)$  is not continuously differentiable. Try the example  $f(x) = |x|$ ,  $a = -1$ ,  $b = 1$ , with different tolerances. The problem is calculator approximation of the derivative with computation of  $y_2 = nDeriv(y_1, x, x)$ . A false derivative value is returned at  $x = 0$ .

Program:MVT

:1->F

:Disp "Y1=F(X)"

:Disp "Y2=DERIV"

:Input "TOLERANCE=",T

flag value

enter f(x) in Y1

enter nDeriv(Y1,X,X) in Y2

start with .05, then try .02, .01 to refine

```

:Input "LEFT END PT=",X
:X->A
:Input "RT END PT=",X
:X->B
:(Y1(B)-Y1(A))/(B-A)->C
:Disp "AVE SLOPE=",C
:A->X
:While X<B
:X+.01->X
:Y2->S
:If abs(S-C)<=T
:Then
:Disp "SOLUTION=",X
:0->F
:End
:End
:If F=1
:Disp "SOLUTION NOT", " FOUND"

```

will take a while to do all computation due to .01

ex 1:  $f(x)=x^3$  (analytic solution= $\pm \frac{1}{\sqrt{3}} = \pm .577$ )

tolerance=.05, left end pt= -1, rt end pt=1

ave slope=1  
solution= -.59  
solution= -.58  
solution= -.57

solution= .57  
solution= .58  
solution= .59

ex 2:  $f(x)=x^3$

tolerance=.02, left end pt= -1, rt end pt=1

ave slope=1  
solution= -.58                      solution= .58

- Price Elasticity of Demand

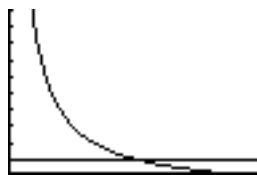
ex:  $p(x) = 5 - 0.03x$                        $\eta = \frac{p/x}{dp/dx}$                       Xmin=0, Xmax=166, Ymin=0, Ymax=10 elastic

when  $|\eta| > 1$ , i.e.  $0 < x \leq 83$  and inelastic when  $84 \leq x \leq 166$

```

Plot1 Plot2 Plot3
\Y1=5-.03X
\Y2=Y1/X
\Y3=nDeriv(Y1,X,
X)
\Y4=abs(Y2/Y3)
\Y5=1
\Y6=

```



## Integral Calculus

- Midpoint Rule

I've used the program MIDPT to motivate the discussion of a Riemann Sum. Input requires the lower and upper limits of integration and number of rectangles to be used. The function to be integrated is entered in  $y_1$  and the user must specify appropriate window settings.

```

Program:MIDPT
:ClrDraw
:Disp "LOWER LIMIT"
:Input A
:Disp "UPPER LIMIT"
:Input B
:Disp "N DIVISIONS"
:Input D
:0-> S
:(B - A)/ D-> W
:1-> J
:Lbl 1
:A + (J - 1) W->L
:L + W/ 2->M
:M->X
:Y1->V
:VW + S->S
:Line (L, 0, L, V)
:L + W->R
:Line (L, V, R, V)
:Line (R, 0, R, V)
:IS > (J, D)
:Goto 1
:Pause
:ClrHome
:Disp "APPROXIMATION="
:Disp S
:Disp "DEF INT="
:fnInt(Y1, X, A, B)->C
:Disp C
:Disp "ABS ERROR = "
:abs (C - S)-> E
:Disp E

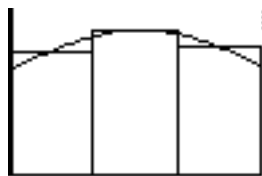
```

enter  $f(x)$  in  $y_1$

will first generate graph of  $f(x)$

$$\text{Distance: } s(t) = \int_a^b |v(t)| dt$$

$$\text{ex: } v = 50 + 20\sin x \quad 0 \leq x \leq 3 \quad N=3$$



```

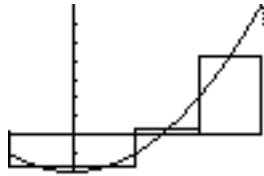
APPROXIMATION=
 191.5078534
DEF INT=
 189.7998499
ABS ERROR=
 1.708003454
      Done

```

An alternate version that allows for function input and automatic window settings is program MIDPT2.

Program:MIDPT2	Automatically set $X_{\min}$ , $X_{\max}$ , $Y_{\min}$ , $Y_{\max}$
:ClrDraw	
:Disp "INPUT YOUR EQN,"	
:Disp " USE QUOTES TO"	store a string of characters in a function
:Disp "START AND END"	
:Disp "Y ="	
:Input Y <sub>1</sub>	
:Disp "LOWER LIMIT"	
:Input A	
:A-> $X_{\min}$	From VARS, WINDOW , Set $X_{\min}$ for graph
:Disp "UPPER LIMIT"	
:Input B	
:B-> $X_{\max}$	Set $X_{\max}$ for graph from VARS, WINDOW
:fMin (Y <sub>1</sub> , X, A, B)->X	From MATH Menu
:Y <sub>1</sub> -> $Y_{\min}$	Evaluates Y <sub>1</sub> at min value X; set $Y_{\min}$ for graph
:fMax ((Y <sub>1</sub> , X, A, B)->X	
:Y <sub>1</sub> -> $Y_{\max}$	Evaluates Y <sub>1</sub> at max value X; set $Y_{\max}$ for graph
:Disp "N DIVISIONS"	
:Input D	
:0->S	
:(B - A)/ D->W	
:1->J	
:Lbl 1	
:A + (J - 1) W->L	
:L + W/ 2->M	
:M->X	
:Y <sub>1</sub> ->V	
:VW + S->S	
:Line (L, 0, L, V)	
:L + W->R	
:Line (L, V, R, V)	
:Line (R, 0, R, V)	
:IS > (J, D)	
:Goto 1	
:Pause	
:ClrHome	
:Disp "APPROXIMATION ="	
:Disp S	
:Disp "DEF INT ="	
: fnInt(Y <sub>1</sub> , X, A, B)-> C	
:Disp C	
:Disp "ABS ERROR = "	
:abs (C - S)->E	
:Disp E	
:ClrDraw	Clear the screen

ex:  $y = x^2 - 2 \quad -1 \leq x \leq 3 \quad N=4$   
 Define  $y_1$  by entering " $x^2 - 2$ " at the prompt



```

APPROXIMATION= 1
DEF INT=
1.333333333
ABS ERROR=
.333333333
Done

```

- Differential Equations

To graph the solution to  $\frac{dy}{dx} = f(x)$  with the boundary condition  $y(A) = B$ , define

$$y_1 = f(x) \quad y_2 = \text{fnInt}(y_1, x, A, x) + B$$

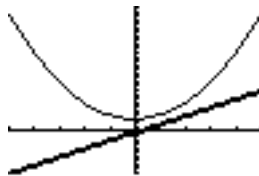
ex:  $y_1 = 2x$

Values for the constants A and B are first stored on the home screen. Window settings are Xmin=-5 Xmax=5 Ymin=-10 Ymax=30 Xres=2.

0→A

3→B

0  
3



Note: With Xres=2, plotting time was approximately 20 seconds. An option is to define the boundary condition constants as  $y_2 = \text{fnInt}(y_1, x, 0, x) + 3$ .

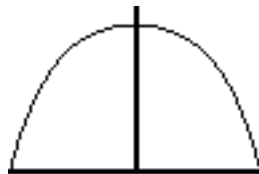
- Arc Length

$$L = \sum_{i=1}^n d_i \quad L = \int_a^b \sqrt{1 + [f'(x)]^2} dx$$

$$y_1 = f(x) \quad y_2 = n\text{Deriv}(y_1, x, x) \quad \text{fnInt}(\sqrt{1 + (y_2)^2}, x, a, b)$$

ex: Gateway Arch in St. Louis

$$y = 693.8597 - 68.7672 \cosh 0.0100333x \quad |x| \leq 299.2239$$



Xmin=-299.2 Xmax=299.2 Ymin=0 Ymax=700 Xres=1

Note: Setting the tolerance to .1 rather than use the default (1E-5) value results in computation time less than 5 secs as opposed to approx 20 secs.

```

fnInt(√(1+(Y2)²), X, -299.2, 299.2, .1)  fnInt(√(1+(Y2)²), X, -299.2, 299.2)
.1)
1479.945323  1479.945319

```

- Arc Length - Parametric Equations

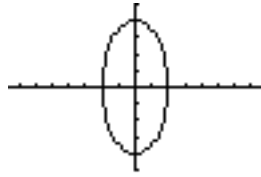


$$L = \int_{t=a}^{t=b} \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2} dt$$

$$x_{1T} = x(t) \quad y_{1T} = y(t) \quad x_{2T} = nDeriv(x_{1T}, T, T)$$

$$y_{2T} = nDeriv(y_{1T}, T, T)$$

ex:  $x(t) = 2 \cos(t)$   $y(t) = 4 \sin(t)$ ,  $0 \leq t \leq 2\pi$



```
fnInt(√((X2T)^2+(
Y2T)^2),T,0,2π,.1
)
19.37879166
```

- Series

I've used the program SERIES to motivate the discussion of convergent and divergent series. Input requires the lower and upper limits of for the sum. The function is entered in  $y_1$ . This routine serves as an alternative to  $sum(seq(y_1, x, begin, end))$  when the list has more than 999 items. Repeated use of  $sum(seq(y_1, x, begin, end))$  with adjusted begin and limits for the series in cases where  $n$  is very large would bypass this problem but would require that intermediate sums be recorded.

Program:SERIES

:Disp "ENTER Y<sub>1</sub> IN X"

:DISP "LOWER LIMIT"

:Input L

:Disp "UPPER LIMIT"

:Input U

:U->X

:0->S

:Lbl 1

:Y<sub>1</sub>+S->S

:DS < (X,L)

:Goto 1

:Disp S

ex:  $\sum_{k=1}^n \left(\frac{1}{3}\right)^k$  geometric series,  $a = r = \frac{1}{3}$

$y_1 = \left(\frac{1}{3}\right)^x$  lower limit=1, upper limit=25, sum= .5

ex:  $\sum_{k=1}^n \frac{1}{k}$  harmonic series

$y_1 = \frac{1}{x}$	lower limit	upper limit	sum
1		100	5.187
	1		500
	1	2000	6.793
			8.178

ex: \$2000 is invested yearly for 30 years and assume interest compounds continuously at 6%. The accumulated amount at the end of the last year is  $\sum_{t=1}^{30} 2000e^{rt}$ .

$$y_1 = 2000e^{0.06x} \quad \text{lower limit}=1, \text{ upper limit}=30, \text{ sum}=\$173,421.72$$

## Multivariable Calculus

- Dot Product

The following program VDOT will compute the dot product of vectors U and V. The orthogonal components of U are (A,B,C) and those of vector V are (D,E,F). Output consists of the magnitude of U, i.e.  $|U|$ , and the magnitude of V, i.e.  $|V|$ . The user will then have to hit ENTER to continue with the output of the dot product

$U \cdot V = AD + BE + CF$ . Finally, the angle between the two vectors is output in degrees using the fact that

$\cos \theta = \frac{U \cdot V}{|U||V|}$ . Three space is assumed, but to use the code for 2D, set the appropriate U or V coefficient to zero.

Program:VDOT

```
:Disp "ENTER(A,B,C)"
```

```
:Input "A=",A
```

```
:Input "B=",B
```

```
:Input "C=",C
```

```
: $\sqrt{A^2 + B^2 + C^2} \rightarrow U$ 
```

```
:Disp "MAGNITUDE OF U=",U
```

```
:Disp "ENTER(D,E,F)"
```

```
:Input "D=",D
```

```
:Input "E=",E
```

```
:Input "F=",F
```

```
: $\sqrt{D^2 + E^2 + F^2} \rightarrow V$ 
```

```
:Disp "MAGNITUDE OF V=",V
```

```
:Pause
```

```
: $AD + BE + CF \rightarrow G$ 
```

```
:Disp "DOT PRODUCT=",G
```

```
: $(360/(2\pi))\cos^{-1}(G/(UV)) \rightarrow T$ 
```

```
:Disp "THETA=",T
```

ex:

$$\begin{cases} \vec{U} = i \\ \vec{V} = j \end{cases} \quad \text{i.e.} \quad \begin{cases} \vec{U} = \langle 1, 0, 0 \rangle \\ \vec{V} = \langle 0, 1, 0 \rangle \end{cases}$$

output is

$$|\vec{U}| = 1, |\vec{V}| = 1, \vec{U} \cdot \vec{V} = 0, \theta = 90$$

ex:

$$\begin{cases} \vec{U} = \langle 2, 3, 1 \rangle \\ \vec{V} = \langle -1, 4, 2 \rangle \end{cases}$$

output is

$$|\vec{U}| = 3.742, |\vec{V}| = 4.583, \vec{U} \cdot \vec{V} = 12, \theta = 45.585$$

- Cross Product

The following program VCROSS will compute the cross product of vectors U and V. The orthogonal components of U in the program are (A,B,C) and those of vector V are (D,E,F). Output consists of the magnitude of U, i.e.  $|U|$ , and the magnitude of V, i.e.  $|V|$ . The user will then have to hit ENTER to continue with the output of the cross product which is defined in the program as  $U \times V = \langle BF - CE, CD - AF, AE - BD \rangle = \langle L, M, N \rangle$ . If we consider the more mathematical representation  $U = \langle u_1, u_2, u_3 \rangle$  and  $V = \langle v_1, v_2, v_3 \rangle$ , then  $U \times V = \langle u_2v_3 - u_3v_2, u_3v_1 - u_1v_3, u_1v_2 - u_2v_1 \rangle$ . The user will then have to hit ENTER to continue with the output of the magnitude of the cross product  $U \times V$ , i.e.  $|U \times V|$ .

```

Program:VCROSS
:Disp "ENTER(A,B,C)
:Input "A=",A
:Input "B=",B
:Input "C=",C
: $\sqrt{A^2 + B^2 + C^2} \rightarrow U$ 
:Disp "MAGNITUDE OF U=",U
:Disp "ENTER(D,E,F)"
:Input "D=",D
:Input "E=",E
:Input "F=",F
: $\sqrt{D^2 + E^2 + F^2} \rightarrow V$ 
:Disp "MAGNITUDE OF V=",V
:Pause
:BF-CE  $\rightarrow$  L
:CD-AF  $\rightarrow$  M
:AE-BD  $\rightarrow$  N
:Disp "CROSS PRODUCT IS"
:Disp "L=",L
:Disp "M=",M
:Disp "N=",N
:Pause
: $\sqrt{L^2 + M^2 + N^2} \rightarrow W$ 
:Disp "MAGNITUDE OF W=",W

```

ex:

$$\begin{cases} \vec{U} = i \\ \vec{V} = j \end{cases} \quad \text{i.e.} \quad \begin{cases} \vec{U} = \langle 1, 0, 0 \rangle \\ \vec{V} = \langle 0, 1, 0 \rangle \end{cases}$$

output is

$$|\vec{U}| = 1, |\vec{V}| = 1, \vec{U} \times \vec{V} = \langle 0, 0, 1 \rangle, |\vec{U} \times \vec{V}| = 1$$

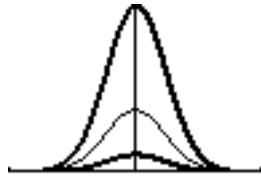
ex: Find a unit vector perpendicular to the plane determined by  $\vec{U} = \langle 2, 3, 1 \rangle$  and  $\vec{V} = \langle 1, -1, 2 \rangle$ .

output is  $|\vec{U}| = 3.742, |\vec{V}| = 2.449, \vec{U} \times \vec{V} = \langle 7, -3, -5 \rangle, |\vec{U} \times \vec{V}| = 9.110$

- Slices

To simulate 3D graphing, slices can be taken for functions of the form  $z = f(x, y)$ .

ex:  $z = e^{-(x^2+y^2)}$ ,  $y = 0, \pm 1, \pm 1.5$



```

Plot1 Plot2 Plot3
Y1= e^(-X^2)
Y2= e^(-(X^2+1))
Y3= e^(-(X^2+1.5^2))
)
)
)
Y4=
Y5=
Y6=

```

An alternate approach to quickly graph  $z = f(x, y)$  slices is to define  $y_1 = e^{-(x^2+A^2)}$ . Then store values as you choose for A at the home screen.

- Partial Derivatives

This program can be used to test for a max or min at the critical point for x and y. The program will compute the partials  $f_x, f_y, f_{xx}, f_{yy}, f_{xy}, f_{yx}$  for the function  $z = f(x, y)$  which is stored in  $Y_1$ . The user must input the critical point values of x and y. The second partials test  $D = f_{xx}f_{yy} - (f_{xy})^2$  is evaluated at the critical point. When  $f_{xx} > 0$  (or  $f_{yy} > 0$ ) and  $D > 0$ , the critical point is a min. When  $f_{xx} < 0$  (or  $f_{yy} < 0$ ) and  $D > 0$ , the critical point is a max. When  $D < 0$ , there is a saddle point at the input value. The program can also be used to find directional derivatives at specified points.

```

Program:PARTIALS
:Disp "STORE Z IN Y1"
:Input "X=",X
:Input "Y=",Y
:nDeriv(Y1,X,X)->A
:nDeriv(Y1,Y,Y)->B
:Disp "FX=",A
:Disp "FY=",B
:Pause
:nDeriv(nDeriv(Y1,X,X),X,X)->C
:nDeriv(nDeriv(Y1,Y,Y),Y,Y)->D
:Disp "FXX=",C
:Disp "FYY=",D
:Pause
:nDeriv(nDeriv(Y1,X,X),Y,Y)->E
:nDeriv(nDeriv(Y1,Y,Y),X,X)->F
:Disp "FGY=",E
:Disp "FYX=",F
:Pause
:Disp "SECOND PARTIALS"
:Disp "TEST"
:Disp "D=",C*D-E*F

```

ex:  $z = e^{-(x^2+y^2)}$

1) At x=0 and y=1, output is

$f_x = 0, f_y = -.736, f_{xx} = -.736, f_{yy} = .736, f_{xy} = f_{yx} = 0$  (Disregard computation for D since this is not a critical point.)

2) At the critical point  $x=0$  and  $y=0$ , output is

$$f_x = 0, f_y = 0, f_{xx} = -2, f_{yy} = -2, f_{xy} = f_{yx} = 0, D = f_{xx}f_{yy} - (f_{xy})^2 = 4$$

- Iterated Integral

This program evaluates the iterated integral  $\int_{y=c}^{y=d} \int_{x=a}^{x=b} f(x,y) dx dy$ . Input consists of limits for x, i.e.  $a \leq x \leq b$ , and limits for y, i.e.  $c \leq y \leq d$ . The number of divisions N represents divisions in the y direction. The width of each increment  $\Delta y$  is  $\frac{d-c}{N}$  and the midpoint of the appropriate interval is stored in Y for use in  $fnInt(y_1, x, a, b)$  computation. The approximate sum representing the value of the iterated integral is computed as  $\sum_{i=1}^N \Delta y_i fnInt(y_1, x, a, b)$  where  $y_1 = f(x, y)$ .

Program:DXDY

:0->V

:Prompt A

:Prompt B

:Prompt C

:Prompt D

:Prompt N

:(D-C)/N->W

:C+W/2->Y

:For(K,1,N)

:V+W\*fnInt(Y1, X, A, B)->V

:Y+W->Y

:End

:Disp V

Guideline: Start with N=5, 10, ... and enter  $z=f(x,y)$  in Y1

$$\text{ex: } \int_{y=-\infty}^{y=\infty} \int_{x=-\infty}^{x=\infty} e^{-(x^2+y^2)} dx dy$$

As an approximation, consider  $\int_{y=-3}^{y=3} \int_{x=-3}^{x=3} e^{-(x^2+y^2)} dx dy$ .

Setting A=-3 B=3 C=-3 D=3 N=5, output is 3.148143999 for the volume.  
When N=10, the volume approximation is 3.14148307.

$$\text{ex: } \int_{y=0}^{y=2} \int_{x=0}^{x=1} (4-x-y) dx dy$$

Setting A=0 B=1 C=0 D=2 N=5, output is 5 for the volume of the solid under the

plane  $z = 4 - x - y$  over the rectangle  $0 \leq x \leq 1, 0 \leq y \leq 2$ .

## Appendix A

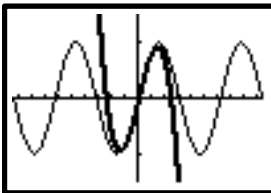
### Polynomial Approximations

After generating a Maclaurin or Taylor series approximation for an appropriate transcendental function, a common technique to examine agreement between the polynomial approximation  $P_n(x)$  and the function is to literally use the Y= Editor and code successive approximations. An alternate and quicker approach is as follows.

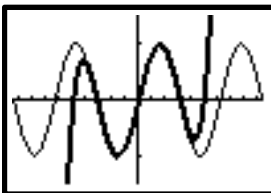
- $\sin(x)$   
For  $P_n(x) = \sum_{k=1}^n (-1)^{k+1} \frac{x^{2k-1}}{(2k-1)!}$ ,  $\sin(x)$  window settings are set as:

```
WINDOW
Xmin=-9.424777...
Xmax=9.4247779...
Xscl=1
Ymin=-1.5
Ymax=1.5
Yscl=1
Xres=1
```

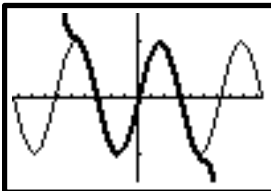
With Xres=1, graphing is slow as the number of terms in the series increases.



```
Plot1 Plot2 Plot3
\Y1=sin(X)
\Y2=sum(seq((-1)^(K+1)*X^(2K-1)/(2K-1)!,K,1,2,1))
\Y3=
\Y4=
```

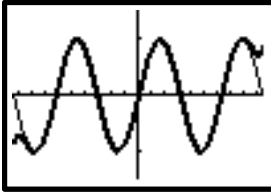


```
Plot1 Plot2 Plot3
\Y1=sin(X)
\Y2=sum(seq((-1)^(K+1)*X^(2K-1)/(2K-1)!,K,1,5,1))
\Y3=
\Y4=
```



```
Plot1 Plot2 Plot3
\Y1=sin(X)
\Y2=sum(seq((-1)^(K+1)*X^(2K-1)/(2K-1)!,K,1,6,1))
\Y3=
\Y4=
```

With Xres=1, the following took more than 2 minutes to graph:



```

Plot1 Plot2 Plot3
\Y1=sin(X)
\Y2=sum(seq((-1)^(K+1)*X^(2K-1)/(2K-1)!,K,1,11,1))
\Y3=
\Y4=

```

Setting Xres=4 reduced the time to 40 seconds without much change in appearance.

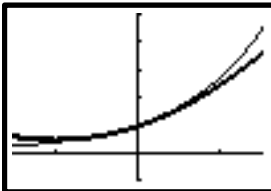
•  $e^x$

For  $P_n(x) = 1 + \sum_{k=1}^n \frac{x^k}{k!}$ , a 2nd order polynomial approximation for  $e^x$  uses the window

```

WINDOW
Xmin=-1.5
Xmax=1.5
Xscl=1
Ymin=-1
Ymax=5
Yscl=1
Xres=1

```



```

Plot1 Plot2 Plot3
\Y1=e^(X)
\Y2=1+sum(seq(X^K/K!,K,1,2,1))
\Y3=
\Y4=
\Y5=
\Y6=

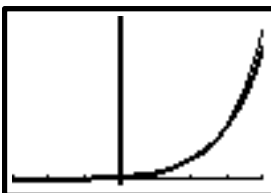
```

Changing the window and resolution enables us to quickly generate a 6<sup>th</sup> order approximation:

```

WINDOW
Xmin=-3
Xmax=4
Xscl=1
Ymin=-2
Ymax=60
Yscl=1
Xres=4

```



```

Plot1 Plot2 Plot3
\Y1=e^(X)
\Y2=1+sum(seq(X^K/K!,K,1,6,1))
\Y3=
\Y4=
\Y5=
\Y6=

```

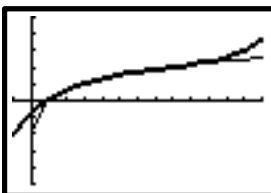
- $\ln(x)$

For  $P_n(x) = \ln(c) + \sum_{k=1}^n (-1)^{k+1} \frac{(x-c)^k}{c^k k}$  approximations to  $\ln(x)$ , the following window for expansions about  $c=5$  was used:

```

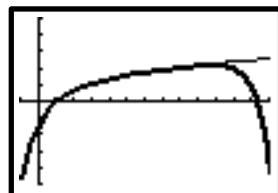
WINDOW
Xmin=-1
Xmax=13
Xscl=1
Ymin=-5
Ymax=5
Yscl=1
Xres=4
  
```

Storing 5 in C, i.e. 5->C generates the following 5th, 10th and 15th order approximations:



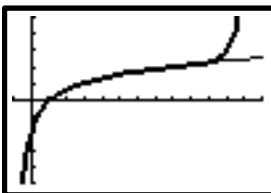
```

Plot1 Plot2 Plot3
\Y1=ln(X)
\Y2=ln(C)+sum(se
a((-1)^(K+1)(X-C
)^K/(C^K)K),K,1
,5))
\Y3=
\Y4=
  
```



```

Plot1 Plot2 Plot3
\Y1=ln(X)
\Y2=ln(C)+sum(se
a((-1)^(K+1)(X-C
)^K/(C^K)K),K,1
,10))
\Y3=
\Y4=
  
```



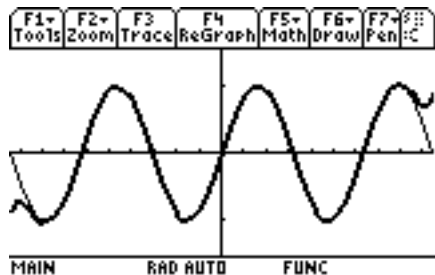
```

Plot1 Plot2 Plot3
\Y1=ln(X)
\Y2=ln(C)+sum(se
a((-1)^(K+1)(X-C
)^K/(C^K)K),K,1
,15))
\Y3=
\Y4=
  
```

- TI-89 Implementation

For  $\sin(x)$ , window settings were the same as those used previously, but the TI-89 default  $Xres=2$  was used. The following polynomial approximation of degree 21 took 35 seconds to graph:



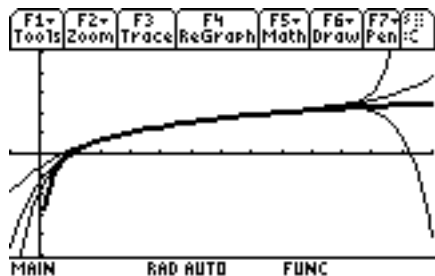


The procedure was to use the **taylor** function and entries were:

1. To generate a polynomial expansion of degree 21 about  $c=0$ , enter **taylor** ( $\sin(x)$ ,  $x$ , 21,0) at the home screen. The key strokes to access the **taylor** function are: F3, option 9. Use the *sin* function key to enter it as the expression to approximate.
2. Store the polynomial expansion in  $y_2$ . The key strokes are: STO- $\rightarrow y_2(x)$  and the display on the home screen will be  $ans(1)\rightarrow y_2(x)$ .
3. Enter  $y_1=\sin(x)$  as a user defined function and change the style for  $y_2$  to *thick*. The key strokes after defining  $y_1$  at the Y= Editor are: highlight the  $y_2$  polynomial expansion, F6 to select *style*, option 4 for *thick*. Then graph the functions.

Setting Xres=4 reduced the time to 20 seconds without much change in appearance.

For  $\ln(x)$ , the window for expansions about  $c=5$  were consistent with those previously used including Xres=4. The following graph took 45 seconds to generate.



The procedure was to use the **taylor** function and store a 5<sup>th</sup> degree polynomial expansion in  $y_2$ , a 10<sup>th</sup> degree polynomial in  $y_3$ , and a 15<sup>th</sup> degree polynomial in  $y_4$ . The  $\ln$  function was stored in  $y_1$ .

The procedure was:

1. **taylor** ( $\ln(x)$ ,  $x$ , 5, 5);  $ans(1)\rightarrow y_2(x)$  to store the results in  $y_2$ .
2. **taylor** ( $\ln(x)$ ,  $x$ , 10, 5);  $ans(1)\rightarrow y_3(x)$  to store the results in  $y_3$ .
3. **taylor** ( $\ln(x)$ ,  $x$ , 15, 5);  $ans(1)\rightarrow y_4(x)$  to store the results in  $y_4$ .

The style was set to *thick* only for  $y_1=\ln(x)$ .