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
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

### Algebra
1. Inequalities ............................................................................................................... 4  
2. Rational Equations ........................................................................................................ 4 

### Precalculus
1. Radical Equations ......................................................................................................... 5  
2. System of Equations ..................................................................................................... 5  
3. Conics ........................................................................................................................ 6  
   Parabola ................................................................................................................. 6  
   Ellipse .................................................................................................................. 8  
   Hyperbola ................................................................................................................ 10 

### Differential Calculus
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 

### Integral Calculus
1. Midpoint Rule ............................................................................................................. 14  
2. Differential Equations ................................................................................................... 16  
3. Arc Length ................................................................................................................ .... 17  
   Arc Length - Parametric Equations ............................................................................... 17  
4. Series .................................................................................................................... 18 

### Multivariable Calculus
1. Dot Product .............................................................................................................. 18  
2. Cross Product ............................................................................................................ 19  
3. Slices .................................................................................................................... 20  
4. Partial Derivatives ....................................................................................................... 21  
5. Iterated Integral ........................................................................................................... 22 

### Appendix A
Polynomial Approximations .................................................................................................... 23  
1. \( \sin(x) \) ................................................................................................................ 23  
2. \( e^x \) ................................................................................................................ 24  
3. \( \ln(x) \) ........................................................................................................... 25  
4. TI-89 Implementation ................................................................................................... 25
Algebra

- **Inequalities**
  - \(|ax + b| \leq c\) \(y_1 = \text{abs}(ax + b) - c\) and examine where \(y_1 \leq 0\)
  - \(|ax + b| \geq c\) \(y_1 = \text{abs}(ax + b) - c\) and examine where \(y_1 \geq 0\)

  **ex:** \(2x - 1 \geq 5\), define \(y_1 = \text{abs}(2x - 1) - 5\) 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:** \(\frac{2}{x} \leq 3\), define \(y_1 = \text{abs}(\frac{2}{x}) - 3\) 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 \(y_1 = \text{abs}(3x + 1) - 2\text{abs}(x - 6)\) and examine the graph

  with a window set as \(\text{Xmin} = -15\), \(\text{Xmax} = 10\), \(\text{Ymin} = -15\), \(\text{Ymax} = 10\) 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: \(y_1 = \frac{5}{x - 3} - \frac{30}{(x^2 - 9)} - 1\)
      - a) Look for zeroes of \(y_1\) (zoom 6)
      - b) Redefine window to \(y_{\text{min}} = -5\) and \(y_{\text{max}} = 5\)
• Analytically: \( \frac{5}{x-3} - \frac{30}{x^2-9} = 1 \)  
\( \text{LCD is (x+3)(x-3)} \)

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

\( 5x + 15 - 30 = x^2 - 9 \)

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 \( X_{\text{min}}=-10, X_{\text{max}}=20, Y_{\text{min}}=-10, Y_{\text{max}}=10. \)

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

\( (x - 5)^2 = x + 7 \)  

and \( x^2 - 10x + 25 = x + 7 \) or \( x^2 - 11x + 18 = 0 \)

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.

\[ AX = B \]

\[ X = A^{-1}B \]

\( 3x + 2y = 7 \)

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) \text{MATRX, 1, } x^{-1} \text{ enter, yields } A^{-1}

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

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

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

\[
\begin{bmatrix}
\frac{1}{13} & \frac{2}{13} & 1 \\
\frac{5}{13} & \frac{-3}{13} & 1
\end{bmatrix}
\]

• 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.

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        X coordinate of the vertex
:1/A->F        Y coordinate of the vertex
:-D/2->H
:C-(B^2)/(4A)->K        Distance from focus to vertex
: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

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

\text{output is: vertex at } x = -1 \text{ } y = 1, \text{ focus at } x = -1 \text{ } y = .5, \text{ 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

\[ x = \frac{1}{2} y^2 + \frac{1}{2} y + \frac{41}{8} \]

output is: vertex at x= 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 N=1
Then
Disp "DIRECTRIX IS 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

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

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

and the major axis axis has length 2a.

Program: ELLIPSE
ClrHome
Disp "AX^2+CY^2+DX" +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
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

$$
\frac{(x-h)^2}{a^2} - \frac{(y-k)^2}{b^2} = 1 \quad \text{if the transverse axis is horizontal}
$$

$$
\frac{(y-k)^2}{a^2} - \frac{(x-h)^2}{b^2} = 1 \quad \text{if the transverse axis is vertical.}
$$

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
```

Output for the standard form of the equation of a hyperbola. 'a' is associated with the transverse axis,
`b' with the conjugate axis, and $c^2 = a^2 + b^2$

```
: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 "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)
```

Exception case to flag a degenerate hyperbola

\[ \text{ex: } 4x^2 - 3y^2 + 8x + 16 = 0 \]

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 \( y_2 = n\text{Deriv}(y_1, x, x) \)
  
  **ex:** \( s = -16t^2 + 80t + 5 \)  
  \( \text{Xmin}=0, \text{Xmax}=5, \text{Ymin}=-80, \text{Ymax}=160 \)  
  Find zero for \( y_2 \), i.e. Intermediate Value Theorem

- **Point of Diminishing Return:** \( f(x) = 4 \left( 1 + \frac{x - 4}{\sqrt{x^2 - 8x + 20}} \right) \)
  
  \( \text{Xmin}=0, \text{Xmax}=12, \text{Ymin}=-4, \text{Ymax}=10, \text{Xres}=2 \)

- \( y_1 = f(x) \)  
  \( y_2 = n\text{Deriv}(y_1, x, x) \)  
  \( y_3 = n\text{Deriv}(y_2, x, x) \)

- **Inflection Point:** \( f(x) = x^3 - 3x^2 - x + 3 \)
  
  \( \text{Xmin}=-5, \text{Xmax}=5, \text{Ymin}=-10, \text{Ymax}=10, \text{Xres}=2 \)

- \( y_1 = f(x) \)  
  \( y_2 = n\text{Deriv}(y_1, x, x) \)  
  \( y_3 = n\text{Deriv}(y_2, x, x) \)

- **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 \( y_2 = n\text{Deriv}(y_1, x, x) \). A false derivative value is returned at \( x=0 \).

**Program:MVT**

```
:1->F flag value
:Disp "Y1=F(X)"
enter f(x) in Y1
:Disp "Y2=DERIV"
enter nDeriv(Y1,X,X) in Y2
```

12
:Input "TOLERANCE=",\( T \) \[\text{start with .05, then try .02, .01 to refine}\]
:Input "LEFT END PT=",\( X \)
:Input "RT END PT=",\( X \)
:X→A
:X→B
:(\( Y_1(B) - Y_1(A) \))/(B-A)→C
:Disp "AVE SLOPE=",\( C \)
:A→X
:While X<B
:X+.01→X     \[\text{will take a while to do all computation due to .01}\]
:Y2→S
:If abs(S-C)<=T
:Then
:Disp "SOLUTION=",\( X \)
:0→F
:End
:End
:If F=1
:Disp "SOLUTION NOT", " FOUND"

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

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
  \[ \eta = \frac{p/x}{dp/dx} \]
  \( \text{Xmin}=0, \text{Xmax}=166, \text{Ymin}=0, \text{Ymax}=10 \) elastic when \( 0 < x \leq 83 \), i.e. \( \eta > 1 \), and inelastic when \( 84 \leq x \leq 166 \)
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
: Y_1 -> V          enter f(x) in $y_1$
: VW + S -> S
: Line (L, 0, L, V) will first generate graph of f(x)
: 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_1, X, A, B) -> C
: Disp C
: Disp "ABS ERROR = "
: abs (C - S) -> E
: Disp E

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

ex: $v = 50 + 20\sin x$   $0 \leq x \leq 3$   N=3

$\int_0^3 (50 + 20\sin x) dt$

Approximation = 191.5078534
Def Int = 192.7998499
Abs Error = 1.38003454
Done
A program that allows for function input and automatic window settings is program MIDPT2.

Program: MIDPT2

 Automatically set X_{\text{min}}, X_{\text{max}}, Y_{\text{min}}, Y_{\text{max}}

 Automatically set X_{\text{min}} for graph

 store a string of characters in a function

 From VARS, WINDOW

 Set X_{\text{max}} for graph from VARS, WINDOW

 From MATH Menu

 Evaluates Y_1 at min value X; set Y_{\text{min}} for graph

 Evaluates Y_1 at max value X; set Y_{\text{max}} for graph

**Example:**

\[ y = x^2 - 2 \quad -1 \leq x \leq 3 \quad N=4 \]

Define \( y_1 \) by entering "\( x^2 - 2 \)" at the prompt
Differential Equations

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

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

Example: \( y_1 \)

Values for the constants \( A \) and \( B \) are first stored on the home screen. Window settings are \( \text{Xmin} = -5 \quad \text{Xmax} = 5 \quad \text{Ymin} = -10 \quad \text{Ymax} = 30 \quad \text{Xres} = 2 \).

Values for the constants \( A \) and \( B \) are first stored on the home screen. Window settings are \( \text{Xmin} = -5 \quad \text{Xmax} = 5 \quad \text{Ymin} = -10 \quad \text{Ymax} = 30 \quad \text{Xres} = 2 \).

\[ y_1 = 2x \]

Note: With \( \text{Xres} = 2 \), 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

\[ L = \sum_{i=1}^{n} d_i \quad L = \int_{a}^{b} \sqrt{1 + \left( f'(x) \right)^2} \, dx \]

Example: Gateway Arch in St. Louis

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

\[ \text{Xmin} = -299.2 \quad \text{Xmax} = 299.2 \quad \text{Ymin} = 0 \quad \text{Ymax} = 700 \quad \text{Xres} = 1 \]

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.

\[ \text{fnInt}((1+y_2)^2, x, 299.2, 299.2, x, -299.2, 299.2) \]

1479.945339 1479.945319

Arc Length - Parametric Equations
\[ L = \int_{a}^{b} \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2} \, dt \]

\[ x_{1T} = x(t) \quad y_{1T} = y(t) \quad x_{2T} = n\text{Deriv}(x_{1T}, T, T) \quad y_{2T} = n\text{Deriv}(y_{1T}, T, T) \]

\[ \text{ex: } x(t) = 2\cos(t) \quad y(t) = 4\sin(t), \ 0 \leq t \leq 2\pi \]

**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 \( \text{sum(seq}(y_1,x,\text{begin},\text{end})) \) when the list has more than 999 items. Repeated use of \( \text{sum(seq}(y_1,x,\text{begin},\text{end})) \) with adjusted begin and limits for the series in cases where as \( n \) is very large would bypass this problem but would require that intermediate sums be recorded.

**Program: SERIES**

:Disp "ENTER Y1 IN X"
:Disp "LOWER LIMIT"
:Input L
:Disp "UPPER LIMIT"
:Input U
:L1->X
:0->S
:Lbl 1
:Y1+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

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100</td>
<td>5.187</td>
</tr>
<tr>
<td>1</td>
<td>500</td>
<td>6.793</td>
</tr>
<tr>
<td>1</td>
<td>2000</td>
<td>8.178</td>
</tr>
</tbody>
</table>
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 \( \sum_{i=1}^{30} 2000e^{rt} \).

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

**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

\[
\begin{align*}
\text{Disp } & \text{"ENTER(A,B,C)"} \\
\text{Input } & \text{"A=",A} \\
\text{Input } & \text{"B=",B} \\
\text{Input } & \text{"C=",C} \\
\sqrt{ & \left(A^2 + B^2 + C^2\right)} \rightarrow U \\
\text{Disp } & \text{"MAGNITUDE OF U=",U} \\
\text{Disp } & \text{"ENTER(D,E,F)"} \\
\text{Input } & \text{"D=",D} \\
\text{Input } & \text{"E=",E} \\
\text{Input } & \text{"F=",F} \\
\sqrt{ & \left(D^2 + E^2 + F^2\right)} \rightarrow V \\
\text{Disp } & \text{"MAGNITUDE OF V=",V} \\
\text{Pause} & \\
AD + & BE + CF \rightarrow G \\
\text{Disp } & \text{"DOT PRODUCT=",G} \\
\left( & \frac{360}{(2\pi)} \right) \cos^{-1} \left( \frac{G}{\|U\| \|V\|} \right) \rightarrow T \\
\text{Disp } & \text{"THETA=",T}
\end{align*}
\]

**ex:**

\[
\begin{align*}
\hat{U} &= i \\
\hat{V} &= j
\end{align*}
\]

\[\text{i.e.} \quad \hat{U} = \langle 1,0,0 \rangle \quad \hat{V} = \langle 0,1,0 \rangle \]

output is

\[
\|\hat{U}\| = 1, \quad |\hat{V}| = 1, \quad \hat{U} \cdot \hat{V} = 0, \quad \theta = 90
\]

**ex:**

\[
\begin{align*}
\hat{U} &= \langle 2,3,1 \rangle \\
\hat{V} &= \langle -1,4,2 \rangle
\end{align*}
\]

output is

\[
\|\hat{U}\| = 3.742, \quad |\hat{V}| = 4.583, \quad \hat{U} \cdot \hat{V} = 12, \quad \theta = 45.585
\]
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 \(U \times V = \{BF - CE, CD - AF, AE - BD\} = \{L, M, N\}\). If we consider the more mathematical representation \(U = \{u_1, u_2, u_3\}\) and \(V = \{v_1, v_2, v_3\}\), then 
\[U \times V = \{u_2v_3 - u_3v_2, u_3v_1 - u_1v_3, u_1v_2 - u_2v_1\}\. 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
:\[A^2 + B^2 + C^2 \rightarrow U\]
:Disp "MAGNITUDE OF U=",U
:Disp "ENTER(D,E,F)"
:Input "D=",D
:Input "E=",E
:Input "F=",F
:\[D^2 + E^2 + F^2 \rightarrow V\]
: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
:\[L^2 + M^2 + N^2 \rightarrow W\]
:Disp "MAGNITUDE OF W=",W

ex:

\[\begin{align*}
\vec{U} &= \hat{i} \\
\vec{V} &= \hat{j}
\end{align*}\]

i.e. \[\begin{align*}
\vec{U} &= <1,0,0> \\
\vec{V} &= <0,1,0>
\end{align*}\]

output is 
\[|\vec{U}| = 1, \ |\vec{V}| = 1, \ \vec{U} \times \vec{V} = <0,0,1>, \ |\vec{U} \times \vec{V}| = 1\]

ex: Find a unit vector perpendicular to the plane determined by \(\vec{U} = <2,3,1>\) and \(\vec{V} = <1,-1,2>\).

output is \[|\vec{U}| = 3.742, \ |\vec{V}| = 2.449, \ \vec{U} \times \vec{V} = <7,-3,-5>, \ |\vec{U} \times \vec{V}| = 9.110\]

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

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

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( Y_1, X, X) -> A
:nDeriv( Y_1, Y, Y) -> B
:Disp "FX=", A
:Disp "FY=", B
:Pause
:nDeriv(nDeriv( Y_1, X, X), X, X) -> C
:nDeriv(nDeriv( Y_1, Y, Y), Y, Y) -> D
:Disp "FXX=", C
:Disp "FYY=", D
:Pause
:nDeriv(nDeriv( Y_1, X, X), Y, Y) -> E
:nDeriv(nDeriv( Y_1, Y, Y), X, X) -> F
:Disp "FXY=", E
:Disp "FYX=", F
:Pause
:Disp "SECOND PARTIALS"
:Disp "TEST"
:Disp "D=", C*D-E*F

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

1) At $x=0$ and $y=1$, output is
\( f_x = 0, \quad f_y = -0.736, \quad f_{xx} = -0.736, \quad f_{yy} = 0.736, \quad f_{xy} = f_{yx} = 0 \)  
(Disregard computation for D since this is not a critical point.)

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

\[ f_x = 0, \quad f_y = 0, \quad f_{xx} = -2, \quad f_{yy} = -2, \quad f_{xy} = f_{yx} = 0, \quad D = f_{xx}f_{yy} - (f_{xy})^2 = 4 \]

**Iterated Integral**

This program evaluates the iterated integral \( \int_{x=a}^{x=b} \int_{y=c}^{y=d} f(x, y) \, dx \, dy \). 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 \( \text{fnInt}(y_1, x, a, b) \) computation. The approximate sum representing the value of the iterated integral is computed as \( \sum_{i=1}^{N} \Delta y_i, \text{fnInt}(y_1, x, a, b) \) where \( y_i = f(x, y) \).

**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 \( N=5, 10, \ldots \) and enter \( z=f(x, y) \) in \( Y_1 \)

\[
\int_{y=-\infty}^{y=\infty} \int_{x=-\infty}^{x=\infty} e^{-\left(x^2 + y^2\right)} \, dx \, dy
\]

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

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

\[
\int_{y=0}^{y=2} \int_{x=1}^{x=1} (4 - x - y) \, dx \, dy
\]

Setting \( A=0 \quad B=1 \quad C=0 \quad D=2 \quad N=5 \), 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:

\[
\begin{align*}
\text{WIN} & \text{DOM} \\
X_{\text{min}} & = -9.424777 \\
X_{\text{max}} & = 9.4247779 \\
X_{\text{sc}1} & = 1 \\
Y_{\text{min}} & = -1.5 \\
Y_{\text{max}} & = 1.5 \\
Y_{\text{sc}1} & = 1 \\
X_{\text{res}} & = 1
\end{align*}
\]

With \( X_{\text{res}}=1 \), graphing is slow as the number of terms in the series increases.

With \( X_{\text{res}}=1 \), the following took more than 2 minutes to graph:
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

Changing the window and resolution enables us to quickly generate a 6th order approximation:
• \(\ln(x)\)

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:

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

• 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 y2. The key strokes are: \(\text{STO} \rightarrow \text{y2}(x)\) and the display on the home screen will be \(\text{ans}(1) \rightarrow \text{y2}(x)\).

3. Enter \(y1=\sin(x)\) as a user defined function and change the style for \(y2\) to `thick`. The key strokes after defining \(y1\) at the Y= Editor are: highlight the \(y2\) 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\(^{\text{th}}\) degree polynomial expansion in y2, a 10\(^{\text{th}}\) degree polynomial in y3, and a 15\(^{\text{th}}\) degree polynomial in y4. The \(\ln\) function was stored in y1.

The procedure was:

1. `taylor` \((\ln(x), x, 5, 5)\); \(\text{ans}(1) \rightarrow \text{y2}(x)\) to store the results in y2.
2. `taylor` \((\ln(x), x, 10, 5)\); \(\text{ans}(1) \rightarrow \text{y3}(x)\) to store the results in y3.
3. `taylor` \((\ln(x), x, 15, 5)\); \(\text{ans}(1) \rightarrow \text{y4}(x)\) to store the results in y4.

The style was set to `thick` only for \(y1=\ln(x)\).