

Your Students Coding TI-Basic and the TI-84 Plus family opens the door!

Give your students a leg up by blending coding into their lessons. Slope, absolute value and more, can be coded by students on the TI-84 Plus family using the easy-to-learn TI-Basic coding language. Students can gain a deeper understanding of the math. Double the STEM benefit of your class! Examples we will present include the absolute value of a number and finding the equation of a line given two points. See how to help students create TI-Basic programs on the TI-84 Plus CE graphing calculator using the easy-to-learn TI-Basic coding language. Support their algorithmic thinking for step-by-step processes. Students gain a deeper understanding of their work and also learn to code, doubling the STEM effect of your lessons.

No knowledge of TI-Basic or programming necessary for this session. We assume you are familiar with pasting from menus on the TI-84 Plus family calculators.

Terrance A. Mankus Ph.D
University of Illinois at Chicago

Margo Lynn Mankus Ph.D
Texas Instruments

T³ International Conference
March 14, 2020
Dallas, Texas

This handout attempts to replicate what we would have experienced in person. We have added some additional resources that also will give you a jump start on using TI-Basic in your classrooms.

Our hope is everyone is safe and healthy.

Table of Contents

Part I: Some targeted resources for TI STEM and TI Codes	2
• STEM resources for middle grades through high school	2
• TI Codes for the TI-84 Plus family	2
• Session Materials	2
Part II: Our class time is so busy now! How and where do I fit this in?	2
Part III: Nuts and Bolts	3
Part IV: Intro Program: Absolute Value	5
Part V: Program: LINE.....	9

Part I: Some targeted resources for TI STEM and TI Codes

- **STEM resources for middle grades through high school**

<https://education.ti.com/en/resources/stem>

- **TI Codes for the TI-84 Plus family**

<https://education.ti.com/en/activities/ti-codes/84/10-minutes>

These short activities walk through how to enter and run TI-Basic programs on your calculator. The sequencing of the activities guide you through the important concepts in coding from controlling the display, using variables, conditional if...then statements, looping for repeating program lines efficiently, and more. We encourage you to step through this experience yourself AND with your students!

- **Session Materials**

- This handout will give you a perspective on how to blend coding into several standard algebra topics, absolute value and finding the slope of a line given two points.
- Each program covered is also provided as a calculator file to load to the calculators.

Part II: Our class time is so busy now! How and where do I fit this in?

Depending how you decide to blend in coding, you can approach this in several ways. *Please don't stop reading here. You'll get some ideas on motivation throughout the handout.* 😊

```
NORMAL FLOAT AUTO REAL RADIAN MP
EDIT MENU: [◀] [▶] [F5]
PROGRAM: ABSOLUTE
:ClrHome
:Disp "FIND ABSOLUTE VALUE
"
:Disp "OF A NUMBER"
:Disp
:Input "YOUR NUMBER IS ",A
:Disp
:Disp "ABSOLUTE VALUE"
```



Use an Existing Program

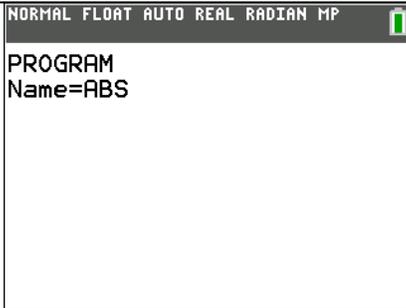
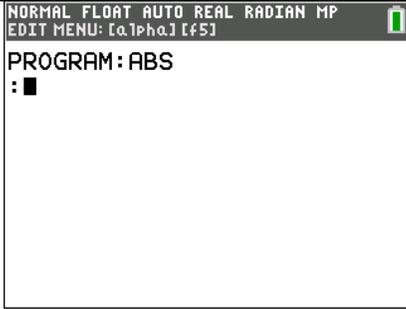
- Read an existing program together. Two are provided for this session. Search for a TI-Basic program on the web. Be specific to a topic such as "slope of a line with two points."
- Play calculator before running the program. Can they figure out what this program will do?
- Execute (Run) the program to see what it does.
- Edit the program to enhance what it does!

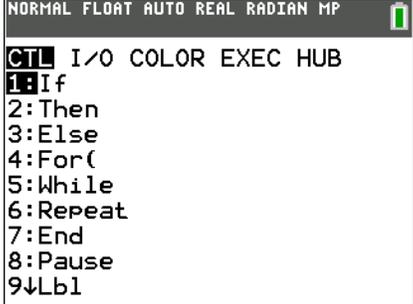
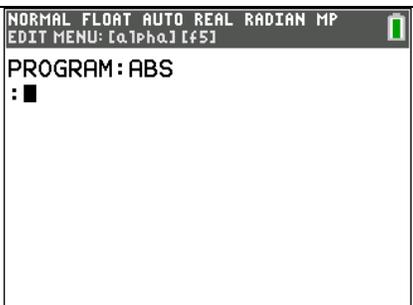
Student Design

- Present a problem to solve or brainstorm with your students about the math "topic of the day."
- Have your students design (story board) and enter their programs.
- Guide them with questions.
- Students execute their program and Edit ("debug") until they feel "it works!"

Part III: Nuts and Bolts

Refer back to this section as needed. Same steps for the TI-84 Plus as well. Try these steps before working on a program. Remember, [2nd] [quit] will get you back to the Home Screen and out of TI-Basic.

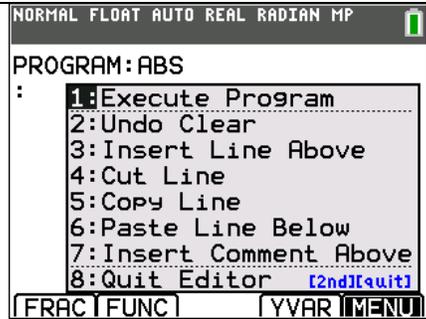
<p>Press [prgm] > NEW</p>	 <p>The screenshot shows the TI-BASIC menu. At the top, it reads 'NORMAL FLOAT AUTO REAL RADIAN MP' and 'TI-BASIC'. Below that, there are three options: 'EXEC EDIT NEW'. The 'NEW' option is highlighted with a cursor. Below the options, it says 'Create New'.</p>
<p>[enter] The cursor is in alpha-lock. Use the keypad to enter the program name. ABS is a fine name. 😊</p>	 <p>The screenshot shows the TI-BASIC menu with 'PROGRAM' selected. Below it, it says 'Name=ABS'. The cursor is positioned at the end of 'ABS'.</p>
<p>[enter] to display the TI-Basic Editor. Each program line will start with a colon.</p>	 <p>The screenshot shows the TI-BASIC Editor. At the top, it reads 'NORMAL FLOAT AUTO REAL RADIAN MP' and 'EDIT MENU: [alpha] [prgm] [f5]'. Below that, it says 'PROGRAM: ABS' and a colon followed by a cursor.</p>

<p>Tips: Press [prgm] while in the Editor to display TI-Basic submenus and commands. [prgm] key double dips here!</p> <p>We will use commands from these two menus.</p> <p>CTL – Control commands for statements such as If/Then which is oh so dear in a math class! Sounds like some Math here! “If $N < 0$, then absolute value is the opposite of N.” <i>We will use N instead of X for the variable. X is used in $Y=$ and changes with the graphing values. So let the calculator savor the use of X for graphing! (Best Practice)</i></p> <p>I/O – Input/Output commands such as Input or Prompt to enter values when your program runs or “Disp” to display output values from the program on the Home Screen! Sounds like more Math! Input value to a variable and display your work!</p>	 <p>NORMAL FLOAT AUTO REAL RADIAN MP</p> <p>CTL I/O COLOR EXEC HUB</p> <p>1: If 2: Then 3: Else 4: For() 5: While 6: Repeat 7: End 8: Pause 9: Lbl</p>  <p>NORMAL FLOAT AUTO REAL RADIAN MP</p> <p>CTL I/O COLOR EXEC HUB</p> <p>1: Input 2: Prompt 3: Disp 4: DispGraph 5: DispTable 6: Output() 7: getKey 8: ClrHome 9: ClrTable</p>
<p>Press [clear] or a menu item number and you’ll return to the Editor.</p>	 <p>NORMAL FLOAT AUTO REAL RADIAN MP</p> <p>EDIT MENU: [alpha] [f5]</p> <p>PROGRAM: ABS</p> <p>: █</p>
<p>How will I enter other lines in a program?</p> <p>Press the familiar keys and menus as needed to paste to the Editor.</p> <p>For the ABS program below, you will need these! [2nd] [test] (above [math]) to paste < Look on the keypad for [sto→] above [on] Look on the keypad for [alpha] [“] above [+]</p>	

How will I execute the program?

On TI-84 Plus CE, from the TI-Basic Editor, press α [f5] (above [graph]). There is a convenient menu as shown! 1:Execute Program

On the TI-84 Plus, [2nd][quit] from the Editor. Then press [prgm] and select your program from the EXEC menu to execute.



Part IV: Intro Program: Absolute Value

Absolute Value of a number is described in many ways. You probably have a favorite such as

“How far is the number from 0?”

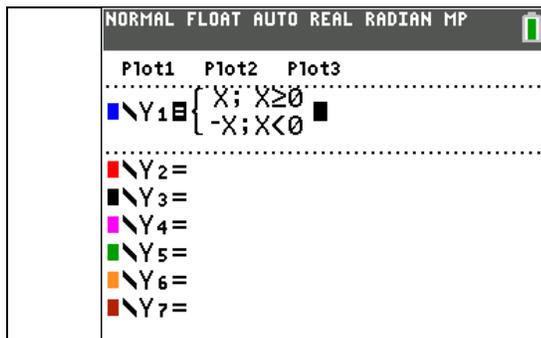
“How many steps would you have to take away from the start?”

“What direction did you walk and how far?”

All of these are great discussions and are needed to put some context around “Why do we even care about the absolute value of a number?” ☺

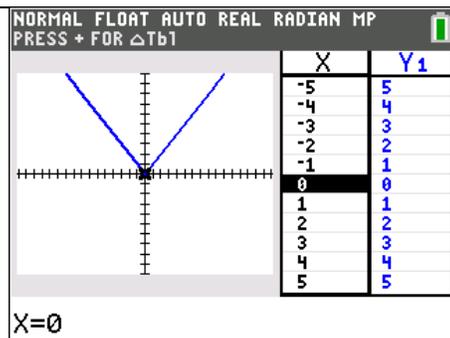
At some point, the student sees a formal definition as a piecewise function shown here using the calculator. **Nice and automatic – right? But how did the calculator find these numbers?**

(Always use the latest OS education.ti.com/84ceupdate.)



[math] MATH B:piecewise(function on the TI-84 Plus CE

Note: We will use N in our program below instead of X for the variable. X is used in Y= and changes with the graphing values. So let the calculator savor the use of X for graphing! (Best Practice)



[mode] GRAPH-TABLE view

Challenge: Let's TELL the calculator HOW to compute the Absolute Value using the language of the calculator, TI-Basic! Forget the automatic answer! Let's be in control of technology!

Remember that you fashion this according to your class time. For a while, you may stay at the "Story Board" stage for some time before entering a program. The "Story Board" stage will seem a lot like how you normally teach!

1. Story board "the absolute value of X"

- Use pair-share or work in groups and talk about the steps they took to write down their answers. Hopefully, some will write down a negative number. ☺ If not, suggest a number for them to write about. This step models how to "story board" the steps of a procedure before starting to code!
- **Questions to ask.** (Fashion this set of questions for YOUR students.)
 - Write down a positive or negative number. (Probing for "What is the **Input?**")
 - As you know, in the definition as shown above, a value for the variable X is requested.
 - What is the absolute value of your number? How did you decide? Listen for "If negative..."
 - Write your answer. (Probing for the "**Display the Output.**")
 - Repeat for another number or teacher can prompt with a number.

2. Vocabulary – This is all we will need!

Warning: Computer Science vocabulary and Math vocabulary sometimes COLLIDE! For this example, only storing to a variable will lead to a discussion!

- **Disp** – "Display" We will want to display text and certainly display the result.
- **Input** – We will have to "tell" the calculator a number. The program will "prompt" for an "Input."
- **→ (Store to a variable)** – We do have to put the number in a variable. Gee – just like in Math classes with a twist! You can change the value of a variable and then store it back to that variable. This is not reflexive. Look on the keypad for [sto→] above [on].
- **If condition** – These are called "conditional" statements. If statements help support logical reasoning.

3. The recipe! What are the steps? Return to their discussion to break this down! Now is the time to speak the calculator language of TI-Basic!

- **Need to input a number to the calculator.** How do we ask for that in code? This code snippet prompts for a value to store in N.

```
Disp "ENTER A NUMBER N"  
Input N
```

- **Need to apply the absolute value definition or "rule" to the number so the calculator knows what to do.** Did the definition lead them to "if $X < 0$, then $-(X)$?" How will we write that in code? Let's use N for the variable. In the calculator, X is the Y= variable. Let's let the calculator grapher have X. ☺ We do want flexible thinkers. This helps!

Here is where \rightarrow is interesting. If N is negative let the opposite of N ($-1*N$) store to N.

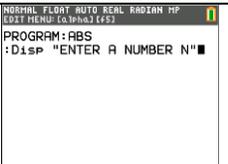
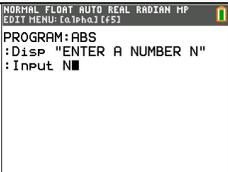
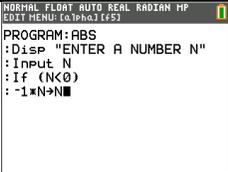
If ($N < 0$)
 $-1*N \rightarrow N$

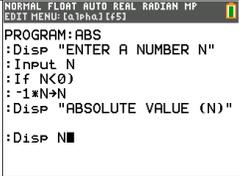
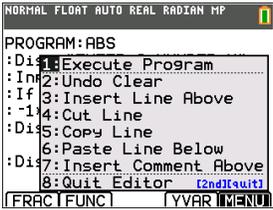
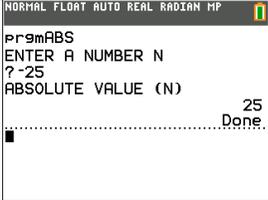
- Tell the calculator to “Disp” the result of the calculation! Make it pretty. Whew!

Disp "ABSOLUTE VALUE (N)"
 Disp N

Here we go! Your final screen will look like this. Please refer to Part III: Nuts and Bolts as needed!

```
NORMAL FLOAT AUTO REAL RADIAN HP
EDIT MENU: Co,PhaJ (f5)
PROGRAM:ABS
:Disp "ENTER A NUMBER N"
:Input N
:If N<0
:-1*N→N
:Disp "ABSOLUTE VALUE (N)"
:Disp N
```

<p>From the Home Screen [prgm]>NEW Enter ABS at Name prompt. [enter]</p>	
<p><i>Program Line:</i> Disp "ENTER A NUMBER N"</p> <p>[prgm] > I/O 3:Disp Use [alpha]["] (above [+]) to enter " Use [2nd] [A-lock] to enter the text between " ".</p>	
<p><i>Program Line:</i> Input N</p> <p>[prgm] > I/O 1:Input Use [alpha] [N] as well.</p>	
<p><i>Program Lines:</i> If ($N < 0$) $-1*N \rightarrow N$</p> <p>[prgm] CTL 1:If Pares are on the keypad above [8] and [9] Press [2nd] [test] (above [math]) for 5:< [(-)] is on the last row of keys by [enter] [sto→] is above [on]</p>	

<p><i>Program Lines:</i> Disp "ABSOLUTE VALUE (N)" Disp N</p> <p>[prgm] > I/O 3:Disp Use [alpha]["] (above [+]) to enter " Use [2nd] [A-lock] to enter the text between " " .</p>	 <p><i>There is no extra line added between the Disp statements. Space is due to the character lineup on the screen.</i></p>
<p>Execute your program!</p> <p>[alpha] [f5] 1:Execute Program</p>	
<p>Here is what you will see!</p> <p>To re-run the program, press [enter] after the program runs.</p>	

Part V: Program: LINE

We offer another program, one that will produce the general equation of a straight line given two points. To construct the program the student will need enter code to test for a variety of conditions.

- Is this a vertical line?
- Is this a horizontal line?
- If neither vertical nor horizontal what is the slope?
- How can the slope be used to generate the coefficients of the x and y variables in the general form of the straight line $AX+BY=C$?
- How can this all be displayed on the calculator? (Advanced commands are used here.)

The calculator file for the program LINE is posted.
Here is the screen display after running the program.

```
NORMAL FLOAT AUTO REAL RADIAN MP
ENTER POINT (X1,Y1)
X1=3
Y1=5
ENTER POINT (X2,Y2)
X2=-2
Y2=-6

11X-5Y=8
SLOPE M =11/5
```

Please use TI Connect CE to send the program to your TI-84 Plus CE or TI-84 Plus. Here is a link to TI Connect CE
<https://education.ti.com/en/products/computer-software/ti-connect-ce-sw>

For more information on TI-Basic programming commands, check out these resources.

TI-Basic Programming Guide:

https://education.ti.com/html/webhelp/EG_TI84PlusCE/EN/index.html

Command and Function Reference Guide:

https://education.ti.com/html/webhelp/EG_TI84PlusCE/EN/index.html

Program LINE is also given here.

```
001 ClrHome
002 Disp "ONLY ENTER INTEGER POINTS"
003 Disp "ENTER POINT (X1,Y1)"
004 Input "X1=",K
005 Input "Y1=",L
006 Disp "ENTER POINT (X2,Y2)"
007 Input "X2=",M
008 Input "Y2=",N
009 Disp " "
010 N-L-A
011 K-M-B
012 NK-LM-C
013 If A=0
014 Then
015 Disp "HORIZONTAL LINE"
016 Disp "Y = "+toString(N)
017 Disp "SLOPE M = 0"
018 Stop
019 End
020 If B=0
021 Then
022 Disp "VERTICAL LINE"
023 Disp "X = "+toString(M)
024 Disp "SLOPE M UNDEFINED"
025 Stop
026 End
027 gcd(abs(A),abs(B)-D
028 gcd(abs(C),D)-D
029 A/D-A
030 B/D-B
031 C/D-C
032 If A<0
033 Then
034 -1*A-A
035 -1*B-B
036 -1*C-C
037 End
038 toString(A)-Str1
039 Str1+"X"-Str1
040 If B<0
041 Then
042 Str1+"-"-Str1
043 Else
044 Str1+"+"-Str1
045 End
046 toString(abs(B))-Str2
047 Str1+Str2+"Y"-Str1
048 toString(C)-Str2
049 Str1+Str2-Str1
050 Disp Str1
051 gcd(abs(A),abs(B)-D
052 A/D-A
053 B/D-B
054 If (abs(B)≠1)
055 Then
056 toString(abs(A))+"/"-Str1
057 Str1+toString(abs(B))-Str1
058 Else
059 toString(abs(A))-Str1
060 End
061 If (A*B>0)
062 "-" + Str1 - Str1
063 Disp "SLOPE M =" + Str1
```