

# Teaching the AC-Method Using a TI-84

Timothy W. Jones

October 16, 2021

## Abstract

Using programs written in TI-Basic increasingly challenging quadratics are factored.

## 1 Introduction

Students learn to factor quadratics of the form  $x^2 + 5x + 6$  using the idea of looking for two number such that when multiplied gives 6 and when added gives 5. We notice that a for loop always can find two such numbers, if they exist. Consider  $x^2 + bx + c$  and

For( $X,1,C$ ), if ( $X+X/C=B$ ), then the two numbers are  $X$  and  $X/C$ . (1)

This plus-plus case is easiest. There are three others:  $x^2 - bx + c$ ,  $x^2 + bx - c$ , and  $x^2 - bx - c$ . Call the four cases  $PP$ ,  $MP$ ,  $PM$ , and  $MM$  as suggested by the (P)lus and (M)inus operations between terms. We will develop a program that can factor these quadratics.

The EASY-FACTORS cases given above have a more complicated cousin. Consider  $10x^2 + 29x + 21$ . With EASY-FACTORS the leading coefficient is 1; here it's not one. The idea for factoring is to multiply the  $a$  and  $c$  coefficient in  $ax^2 + bx + c$  and then as before look for two numbers, say  $d$  and  $e$  such that  $d \cdot e = ac$  and  $d + e = b$ . The EASY-CASE really uses the same strategy -  $a$  is just one, giving just  $c$ . With just a slight modification of (1) these HARD-FACTORS have a strategy:

For( $X,1,AC$ ), if ( $X+AC/X=B$ ), then the two numbers are  $X$  and  $AC/X$ ,  
(2)

where  $A$  and  $C$  are the first and last coefficients of the quadratic. As with EASY-FACTORS, HARD-FACTORS have four cases.

This strategy for solving HARD-FACTORS is called, by some textbooks and teachers, the AC-Method. The method finds the two numbers mentioned and then uses them to express the  $bx$  term. The procedure continues;  $10x^2 + 29x + 21$  is re-expressed using  $d$  and  $e$ :

$$10x^2 + 15x + 14x + 21,$$

where  $d = 15$  and  $e = 14$ . That is, using the algorithm (2),  $d = X = 14$  and  $e = AC/X = 210/14 = 15$ . The final steps in the procedure are to group the first and last two terms, pull out the GCF for  $5x(2x + 3) + 7(2x + 3)$  and then pull out another binomial GCF for  $(2x + 3)(5x + 7)$ , the desired factored form.

The EASY-FACTORS cases provide a good training ground for the HARD-FACTORS cases. The progression shows students problem solving with technology as well as how to parse out cases and look for repeating patterns, not to mention how to look for calculator functions and use them. They are rewarded, I suggest, with good understanding of factoring using the AC-method as well as an ability to do problems easily, accurately, and quickly using technology they made: a program in a calculator.

Details follow.

## The core

The core of EASY-FACTORS and HARD-FACTORS is finding  $d$  and  $e$  such that  $d \cdot e = a \cdot c$  and  $d + e = b$ . The code to accomplish this is given in Figure 1.

```

VAR NAME: ACCORE
-----
001 Prompt A,B,C
002 AC-D
003 For(X,1,abs(D),1)
004 If ((X+D/X=B) or (-X-D/X=B) or (X-D/X=B) or (-X+D/X=B))
005 Then
006 Disp X,D/X
007 End
008 End

```

Figure 1: Core code for both hard and easy factors.

This program allows students to immediately get such  $d$ s and  $e$ s and adjust the signs to fit the middle term. For large coefficients this is the main difficulty in factoring. Try factoring  $90x^2 - 23x - 15$ , for example. With the AC-method one has to multiply out, then factor  $90 \cdot 15 = 1350$  and then find the  $d$  and  $e$  that works. Running the program gives such  $d$  and  $e$ ; one still has to figure out the right signs, not hard: see Figure 2.

```
A=?90
B=? -23
C=? -15
```

```
27
-50
50
-27
```

Figure 2: A difficult factoring problem made much easier.

This ACCORE program is a workhorse; it immediately solves EASYFACTORS and gives the hardest step for HARDFACTORS. To continue to get the calculator to give the complete factors is more work. That's the rest of this paper. For those not wanting to get into these weeds, you might want to use the example quadratics that occur below as test beds for the ACCORE program's accuracy.

## EASY-FACTORS

A few added lines gives the PP case for EASYFACTORS, see Figure 3. This

```
001 Prompt B,C
002 If ((B>0) and (C>0))
003 Then
004 For(X,1,C,1)
005 If (X+C/X=B)
006 Then
007 ClrHome
008 "(x+"+toString(X)+")"->Str1
009 "(x+"+toString(C/X)+")"->Str2
010 Output(1,1,Str1+Str2)
011 Stop
012 End
013 End
014 End
```

Figure 3: Code for EASYFACTORS PP case.

1	$x^2+11x+30$	$x+6$	$x+5$
2	$x^2+11x+18$	$x+2$	$x+9$
3	$x^2+13x+30$	$x+10$	$x+3$
4	$x^2+16x+60$	$x+6$	$x+10$
5	$x^2+14x+48$	$x+6$	$x+8$
6	$x^2+13x+42$	$x+6$	$x+7$
7	$x^2+14x+48$	$x+8$	$x+6$

Figure 4: Testing examples for EASYFACTORS PP case.

code can be tested with quadratic examples given in Figure 4. Code for the EASYFACTORS MP case, the next hardest, is given in Figure 5 with test examples given in Figure 6.

```

015 If ((B<0) and (C>0))
016 Then
017 For(X,1,C,1)
018 If (-X-C/X=B)
019 Then
020 C1rHome
021 "(x-"+toString(X)+")"->Str1
022 "(x-"+toString(C/X)+")"->Str2
023 Output(1,1,Str1+Str2)
024 Stop
025 End
026 End
027 End

```

Figure 5: Code for EASYFACTORS MP case.

8	$x^2-12x+36$	$x-6$	$x-6$
9	$x^2-13x+30$	$x-10$	$x-3$
10	$x^2-12x+32$	$x-8$	$x-4$
11	$x^2-15x+54$	$x-9$	$x-6$
12	$x^2-15x+56$	$x-8$	$x-7$
13	$x^2-15x+56$	$x-8$	$x-7$
14	$x^2-9x+14$	$x-2$	$x-7$

Figure 6: Testing examples for EASYFACTORS MP case.

And so on: Figure 7 gives the code for the remaining PM and MM EASYFACTORS cases and Figure 8 gives more test examples.

```

028 If ((B>0) and (C<0))
029 Then
030 For(X,1,abs(C),1)
031 If (X-abs(C)/X=B)
032 Then
033 ClrHome
034 "(x"+toString(X)+")"->Str1
035 "(x-"+toString(abs(C)/X)+")"->Str2
036 Output(1,1,Str1+Str2)
037 Stop
038 End
039 End
040 End
041 If ((B<0) and (C<0))
042 Then
043 For(X,1,abs(C),1)
044 If (-X+abs(C)/X=B)
045 Then
046 ClrHome
047 "(x-"+toString(X)+")"->Str1
048 "(x"+toString(abs(C)/X)+")"->Str2
049 Output(1,1,Str1+Str2)
050 Stop
051 End
052 End
053 End

```

Figure 7: Code for EASYFACTORS PM and MM cases.

These screen captures for the code used TI's Connect free software that works with the latest TI-84 calculators. The great advantage of using this software is the full range of editing functions available; you can copy and paste the parts of the code that repeats in each of the cases (a lot). You also get a help system for the commands and a way to find the commands available in a systematic way. Finally, with a ctrl-e your code is ported to

15	$x^2-2x-24$	$x-6$	$x+4$
16	$x^2-36$	$x-6$	$x+6$
17	$x^2-2x-35$	$x-7$	$x+5$
18	$x^2-49$	$x-7$	$x+7$
19	$x^2+4x-32$	$x-4$	$x+8$
20	$x^2+8x-20$	$x-2$	$x+10$
21	$x^2+2x-63$	$x-7$	$x+9$
22	$x^2+2x-48$	$x+8$	$x-6$
23	$x^2+8x-20$	$x+10$	$x-2$
24	$x^2+3x-54$	$x+9$	$x-6$
25	$x^2+4x-60$	$x+10$	$x-6$
26	$x^2-8x-20$	$x+2$	$x-10$
27	$x^2-5x-36$	$x+4$	$x-9$
28	$x^2-64$	$x+8$	$x-8$

Figure 8: Testing examples for EASYFACTORS PM and MM cases.

your physical calculator. There it can be tested.

## HARD-FACTORS Structure

Things get a little bit more complicated when coding the HARD-FACTOR cases. With EASY-FACTORS the two numbers  $X$  and  $C/X$  directly give the  $d$  and  $e$  in  $(x \pm d)(x \pm e)$ ;  $d = X$  and  $e = C/X$ . With HARD-FACTORS, as was anticipated in the introduction, there are more steps. You need to pull out GCDs. The puzzling out of what works can be a bit of a trail and error ordeal, but I think that is a good character building aspect of this way of teaching.

This trial and error aspect of building the code is greatly speeded up by having a menu system that allows the different cases to be tried one after the other. Figure 9 gives the screen shot of the code for such systematic testing with pre-loaded quadratics. The code allowing such case studies is given in Figure 10. General input of arbitrary quadratics is given with the OTHER menu option; the code is given in 11.

```
ACFACT
1:PLUSPLUS
2:MINUSPLUS
3:MINUSMINUS
4:PLUSMINUS
5:OTHER
```

Figure 9: Home page screen shot of menu idea.

```
001 Menu ("ACFACT", "PLUSPLUS", U, "MINUSPLUS", S, "MINUSMINUS", T, "PLUSMINUS", V, "OTHER", W)
002 Lbl U
003 8-A
004 22-B
005 15-C
006 prgmMYPP
007 Lbl S
008 8-A
009 -22-B
010 15-C
011 prgmMYMP
012 Lbl T
013 8-A
014 -2-B
015 15-C
016 prgmMYMM
017 Lbl V
018 8-A
019 2-B
020 -15-C
021 prgmMYPM
```

Figure 10: Code for menu program ACFACT; pre-loaded test cases speed debugging.

```

022 Lbl W
023 Prompt A,B,C
024 If ((AC>0) and (B>0))
025 Then
026 prgmMYPP
027 End
028 If ((AC>0) and (B<0))
029 Then
030 prgmMYMP
031 End
032 If ((AC<0) and (B>0))
033 Then
034 prgmMYPM
035 End
036 If ((AC<0) and (B<0))
037 Then
038 prgmMYMM
039 End
040
041 Stop

```

Figure 11: Code continuation for ACFACT. This allows for general inputs.

## HARD-FACTORS Cases

Connect allows several separate files to be available with a mouse click. This makes copy and pasting especially easy. The code for the HARDFACTORS PP case is given in Figure 12; MP Figure 13; PM Figure 14; and MM Figure 15.



MYACMENU.8xp	MYPP.8xp ×	MYMP.8xp	
--------------	------------	----------	--

VAR NAME:

```

001 abs(A)→D
002 abs(B)→E
003 abs(C)→F
004 For(X,1,DF,1)
005 If (DF/X+X=B)
006 Then
007 DF/X→M
008 X→N
009 Disp M,N
010 Disp gcd(D,DF/X),gcd(X,F)
011 Disp D/gcd(D,DF/X),F/gcd(X,F)
012 Pause
013 CTrHome
014 "("+toString(gcd(D,DF/X))+X"+toString(gcd(X,F))+"")→Str1
015 "("+toString(D/gcd(D,DF/X))+X"+toString(F/gcd(X,F))+"")→Str2
016 Output(1,1,Str1+Str2)
017 Stop
018 End
019 End

```

Figure 12: Code for HARDFACTORS PP case.

MYACMENU.8xp	MYPP.8xp	MYMP.8xp ×	MYPM.8xp	MYMM.8xp
--------------	----------	------------	----------	----------

VAR NAME:

```

001 abs(A)→D
002 abs(B)→E
003 abs(C)→F
004 For(X,1,DF,1)
005 If (-DF/X-X=B)
006 Then
007 -DF/X→M
008 -X→N
009 Disp M,N
010 Disp gcd(D,DF/X),gcd(X,F)
011 Disp D/gcd(D,DF/X),F/gcd(X,F)
012 Pause
013 CTrHome
014 "("+toString(gcd(D,DF/X))+X- "+toString(gcd(X,F))+"")→Str1
015 "("+toString(D/gcd(D,DF/X))+X- "+toString(F/gcd(X,F))+"")→Str2
016 Output(1,1,Str1+Str2)
017 Stop
018 End
019 End

```

Figure 13: Code for HARDFACTORS MP case.

MYACMENU.8xp	MYPP.8xp	MYMP.8xp	MYPM.8xp ×	MYMM.8xp
--------------	----------	----------	------------	----------

VAR NAME:

---

```

001  abs(A)→D
002  abs(B)→E
003  abs(C)→F
004  For(X,1,DF,1)
005  If (-DF/X+X=B)
006  Then
007  -DF/X→M
008  X→N
009  Disp M,N
010  Disp gcd(D,DF/X),gcd(X,F)
011  Disp D/gcd(D,DF/X),F/gcd(X,F)
012  Pause
013  CTrHome
014  "("+toString(gcd(D,DF/X))+X"+toString(gcd(X,F))+")"-Str1
015  "("+toString(D/gcd(D,DF/X))+X- "+toString(F/gcd(X,F))+")"-Str2
016  Output(1,1,Str1+Str2)
017  Stop
018  End
019  End

```

Figure 14: Code for HARDFACTORS PM case.

MYACMENU.8xp	MYPP.8xp	MYMP.8xp	MYPM.8xp	MYMM.8xp ×
--------------	----------	----------	----------	------------

VAR NAME:

---

```

001  abs(A)→D
002  abs(B)→E
003  abs(C)→F
004  For(X,1,DF,1)
005  If (DF/X-X=B)
006  Then
007  DF/X→M
008  -X→N
009  Disp M,N
010  Disp gcd(D,DF/X),gcd(X,F)
011  Disp D/gcd(D,DF/X),F/gcd(X,F)
012  Pause
013  CTrHome
014  "("+toString(gcd(D,DF/X))+X- "+toString(gcd(X,F))+")"-Str1
015  "("+toString(D/gcd(D,DF/X))+X"+toString(F/gcd(X,F))+")"-Str2
016  Output(1,1,Str1+Str2)
017  Stop
018  End
019  End

```

Figure 15: Code for HARDFACTORS MM case.

1	$9x^2+12x+4$		$3x+2$	$3x+2$
2	$35x^2+32x+5$		$5x+1$	$7x+5$
3	$1x^2+3x+2$		$1x+2$	$1x+1$
4	$8x^2+15x+7$		$8x+7$	$1x+1$
5	$6x^2+5x+1$		$2x+1$	$3x+1$
6	$12x^2+47x+45$		$3x+5$	$4x+9$
7	$21x^2+53x+30$		$3x+5$	$7x+6$
8	$32x^2+84x+27$		$4x+9$	$8x+3$
9	$8x^2+30x+7$		$2x+7$	$4x+1$
10	$14x^2+29x+12$		$2x+3$	$7x+4$
11	$9x^2-30x+16$		$3x-8$	$3x-2$
12	$25x^2-55x+18$		$5x-2$	$5x-9$
13	$3x^2-7x+4$		$1x-1$	$3x-4$
14	$1x^2-8x+15$		$1x-5$	$1x-3$
15	$9x^2-43x+28$		$1x-4$	$9x-7$
16	$18x^2-33x+14$		$3x-2$	$6x-7$
17	$32x^2-76x+35$		$4x-7$	$8x-5$
18	$12x^2-17x+6$		$4x-3$	$3x-2$
19	$3x^2-5x+2$		$3x-2$	$1x-1$
20	$10x^2-41x+21$		$5x-3$	$2x-7$
21	$9x^2-14x-8$		$1x-2$	$9x+4$
22	$8x^2-6x-5$		$4x-5$	$2x+1$
23	$3x^2-1x-4$		$3x-4$	$1x+1$
24	$20x^2+44x-15$		$10x-3$	$2x+5$
25	$40x^2-29x-56$		$5x-8$	$8x+7$
26	$4x^2-20x-1$		$2x-1$	$2x+1$
27	$32x^2-12x-27$		$8x-9$	$4x+3$
28	$6x^2-5x-4$		$3x-4$	$2x+1$
29	$14x^2-3x-5$		$7x-5$	$2x+1$
30	$24x^2+5x-36$		$8x-9$	$3x+4$

Figure 16: Test examples for HARDFACTORS.

## Conclusion

Notice how the ACCORE if statement with its three *ors* parses into both EASYFACTORS and HARDFACTORS program solutions.

We can demonstrate the nice features of Connect by answering the reader's request to see the program in action. Choosing other from the menu and inputting the twenty-first example quadratic, we can capture the screen from the attached calculator using Connect. This feature is shown in Figure 17. The result of the calculator screen capture is shown in Figure 18.

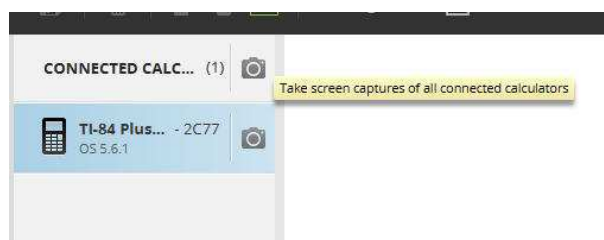


Figure 17: Connect allows screen captures of attached calculators.

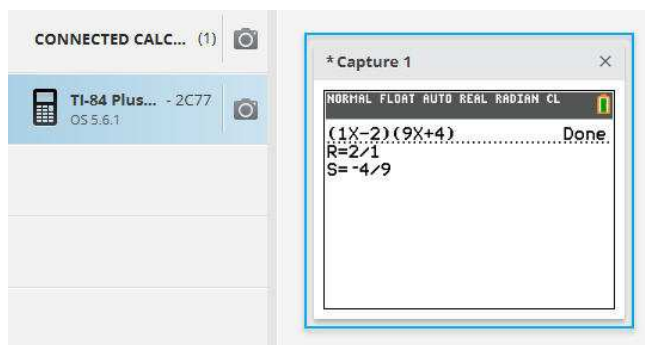


Figure 18: Output for example 21 quadratic with roots added.

After creating these programs in Connect and porting them to an attached physical calculator, there is the question of uploading the program to Smartview's emulator calculator. After saving the program file in Connect, open Smartview and navigate as shown in Figure 19. We've uploaded the code for the MM case; Figure 20 shows the added code for giving the roots, as demoed with the 21st example quadratic. Note that this code is cut off;

the tiny screens of the emulator can't show more than a few lines. We go back to Connect and see all the code well captured in Figure 21. Connect, unlike Smartview, has larger, adjustable screens.

The ideal, somebody tell TI, is to connect Connect with Smartview, so you don't have to upload files to it as a separate step. Still using a doc cam one can create code in Connect, hit the ctrl+e to port to an attached physical calculator, and show that physical calculator running the program, albeit with a necessary switch from computer to doc cam mode for the white board.

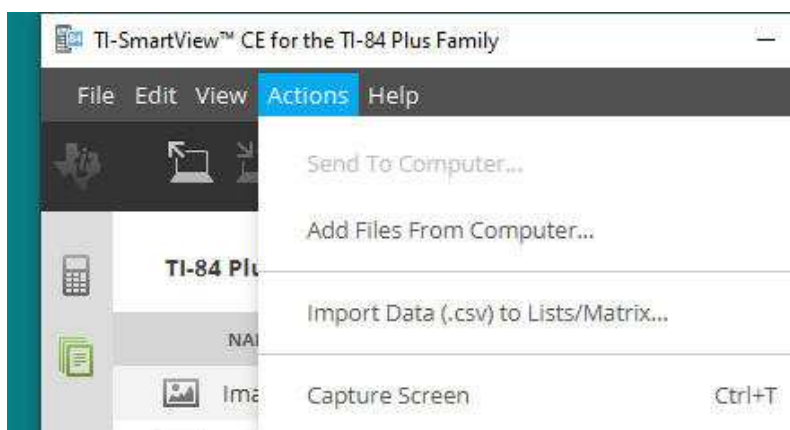


Figure 19: Open Smartview, go to the bottom icon on the left, use the menu to load a file from the computer (from Connect).

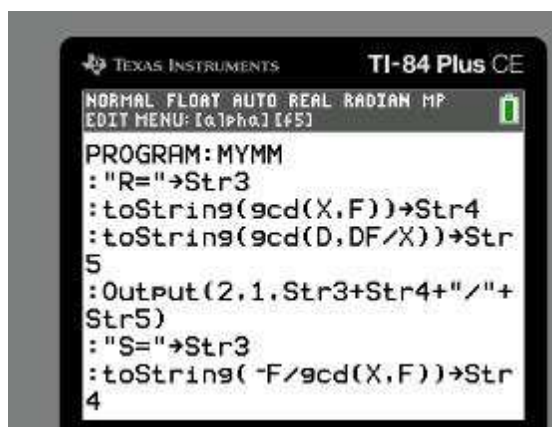


Figure 20: Code added to the MM case to show roots.

```

017 "R="→Str3
018 toString(gcd(X,F))→Str4
019 toString(gcd(D,DF/X))→Str5
020 Output(2,1,Str3+Str4+"/"+Str5)
021 "S="→Str3
022 toString(-F/gcd(X,F))→Str4
023 toString(D/gcd(D,DF/X))→Str5
024 Output(3,1,Str3+Str4+"/"+Str5)
025 Output(4,1,"")

```

Figure 21: Connect allows for better classroom displays of line numbered code in adjustable sizes.