

# Programming the Casio fx-9750/9860 FACTSUB

Note Title

2014-08-08

This is a subroutine that will be used by some of our other programs. This calculates all factors of a number and stores them in List 1. As you know factoring is one of our 3 main tools to solve problems. So this is a handy tool.

To find all factors of  $n$ , we simply check all the numbers from 1 to  $\sqrt{n}$  and if it is divisible, then it is a factor. To calculate the other half of the factor pair, we use the quotient. If  $n$  is a perfect square, we don't make  $\sqrt{n}$  a factor pair.

```

0 =====FACTSUB =====
1 √N→R#
2 (<1)→List 1#
3 I→I#
4 For 2→A To R#
5 N÷A→Q#
6 If Frac Q=0#
7 Then I+1→I#
8 A→List 1[I]#
9 IfEnd#
10 Next#
11 If Frac R=0#
12 Then I-1→J#
13 Else I→J#
14 IfEnd#
15 For J→A To 1 Step -1#
16 I+1→I#
17 N÷List 1[A]→List 1[I]
18 #
19 Next#
20 TOP|BTM|SRC|MENU|A↔3|CHAR
    
```

```

0 MENU PRGM EXE F3 tan X,θ,T ln ÷ × 1 log EXE
1 SHIFT x² ALPHA 8 → ALPHA 6 EXE
2 SHIFT × 1 SHIFT ÷ → SHIFT 1 1 EXE
3 1 → ALPHA ( EXE
4 SHIFT VARS F1 F6 F1 2 → ALPHA X,θ,T F2 ALPHA 6 EXE
5 ALPHA 8 ÷ ALPHA X,θ,T → ALPHA 5 EXE
6 SHIFT VARS F1 F1 OPTN F6 F4 F3 ALPHA 5 SHIFT VARS F6 F3 F1
  0 EXE
7 SHIFT VARS F1 F2 ALPHA ( + 1 → ALPHA ( EXE
8 ALPHA X,θ,T → SHIFT 1 1 SHIFT + ALPHA ( SHIFT - EXE
9 SHIFT VARS F1 F4 EXE
10 SHIFT VARS F1 F6 F4 EXE
11 SHIFT VARS F1 F1 OPTN F6 F4 F3 ALPHA 6 SHIFT VARS F6 F3 F1
  0 EXE
12 SHIFT VARS F1 F2 ALPHA ( - 1 → ALPHA ) EXE
13 SHIFT VARS F1 F3 ALPHA ( → ALPHA ) EXE
14 SHIFT VARS F1 F4 EXE
15 SHIFT VARS F1 F6 F1 ALPHA ) → ALPHA X,θ,T F2 1 F3 (-) 1
  EXE
16 ALPHA ( + 1 → ALPHA ( EXE
17 ALPHA 8 ÷ SHIFT 1 1 SHIFT + ALPHA X,θ,T SHIFT - → SHIFT 1
  1 SHIFT + ALPHA ( SHIFT - EXE
18 SHIFT VARS F1 F6 F4 EXE
    
```

We need a very short program to test our subprogram.

```

0 =====FACTORS =====
1 "N"?+N#
2 Prog "FACTSUB"
3 List 1
4 "NUM FACTORS":I#
  
```

TOP BTM SRC MENU A↔B CHAB



Test the following values to make sure everything is entered correctly.

N?  
9  
Done  
NUM FACTORS  
3

ANS  
1 [ ]  
2 [ 3 ]  
3 [ 9 ]

1

N?  
12  
Done  
NUM FACTORS  
6

ANS  
1 [ ]  
2 [ 2 ]  
3 [ 3 ]  
4 [ 4 ]  
5 [ 6 ]

1

N?  
7  
Done  
NUM FACTORS  
2

ANS  
1 [ ]  
2 [ 7 ]

1

Here is another short program that uses factoring. More specifically, it looks at the number of factors. If it is 2, then it is prime. If it is odd, then it is a perfect square. Otherwise the number is not prime. Although it looks long, most of it is text so there is not much to debug.

```

0 =====ISPRIME =====
1 "N"?→N#
2 Prog "FACTSUB"#
3 If I=2#
4 Then " IS PRIME"#
5 Else If Frac (I÷2)≠0#
6 Then " IS A PERFECT S
  QUARE"#
7 Else " IS NOT PRIME"#
8 IfEnd:IfEnd#
TOP |BTM|SRC|MENU|A↔B|CHAR

```



Test the following values to make sure everything is entered correctly.

```

N?
13
IS PRIME

```

```

N?
81
IS A PERFECT SQUARE

```

```

N?
72
IS NOT PRIME

```