

Counting Factors

Answers

7 8 9 10 11 12



TI-Nspire



Coding



Student



60 min

Finding Factors

There are many ways to determine the quantity of factors for a specified number. The most common method is to test the divisibility for every number up to the specified number.

Example: Determine the quantity of factors for the number 18.

| | | | | | |
|---------------------------------|--------------|---------------------------------|--------------|---------------------------------|--------------|
| $18 \div 1 = 18$ | Factor | $18 \div 2 = 9$ | Factor | $18 \div 3 = 6$ | Factor |
| $18 \div 4 = 4 \text{ rem } 2$ | Not a factor | $18 \div 5 = 3 \text{ rem } 3$ | Not a factor | $18 \div 6 = 3$ | Factor |
| $18 \div 7 = 2 \text{ rem } 4$ | Not a factor | $18 \div 8 = 2 \text{ rem } 2$ | Not a factor | $18 \div 9 = 2$ | Factor |
| $18 \div 10 = 1 \text{ rem } 8$ | Not a factor | $18 \div 11 = 1 \text{ rem } 7$ | Not a factor | $18 \div 12 = 1 \text{ rem } 6$ | Not a factor |
| $18 \div 11 = 1 \text{ rem } 7$ | Not a factor | $18 \div 12 = 1 \text{ rem } 6$ | Not a factor | $18 \div 13 = 1 \text{ rem } 5$ | Not a factor |
| $18 \div 14 = 1 \text{ rem } 4$ | Not a factor | $18 \div 15 = 1 \text{ rem } 3$ | Not a factor | $18 \div 16 = 1 \text{ rem } 2$ | Not a factor |
| $18 \div 17 = 1 \text{ rem } 1$ | Not a factor | $18 \div 18 = 1$ | Factor | | |

The process above is exhaustive and you may already have ideas on how this can be sped up, however this concept will provide a basis for a simple program to count the quantity of factors of a given number.

Teacher Note:

- This activity is designed as a companion to the “Factors that Count” activity on the Texas Instruments site – Australian Nspired Curriculum where students use inquiry based techniques to find a way to calculate the quantity of factors for any given number based on its prime factorisation. The focus of the activity is on counting factors, hence this program counts factors, but does not actually display them. Suggestions on how to display them are included in this teacher notes and answers.
- There is a follow up activity to this one titled: “Fast Facts” which explores how this program can be written more efficiently and also to display the actual factors themselves for the purposes of ‘debugging’. The Fast Facts can produce results 100’s of times faster, depending on the numbers chosen ... 1000’s of times faster!

Instructions:

Start a new document and insert a calculator application.

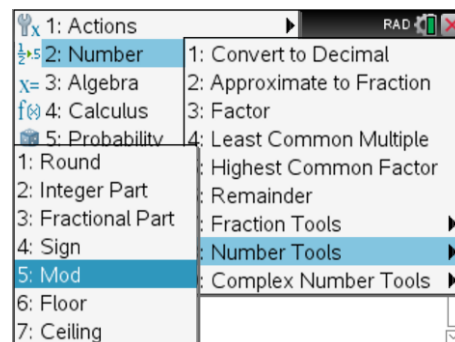
Locate the **mod** command using: **Number > Number Tools > Mod**

Determine the result of the following calculations:

Mod(18,6)

Mod(18,5)

Mod(18,12)



Question: 1.

Based on your experimentation, what value does the MOD command return?

The mod command returns the remainder upon division.

Question: 2.

If $\text{MOD}(a, b) = 0$, what does this say about the relationship between a and b ?

If the 'remainder' of $a \div b = 0$ then b must be a factor of a .

Question: 3.

If $\text{MOD}(a, 2) = 0$, what does this say about the value of a ?

When a number is divided by 2 and the 'remainder' is 0 then the number must be even.

Question: 4.

Write a statement similar to those above that would be true for any odd number ' a '.

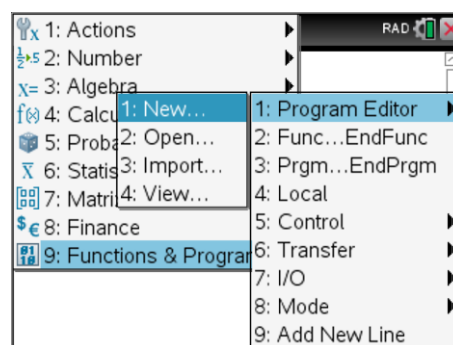
$\text{mod}(a, 2) = 1$... when an odd number is divided by 2 the remainder will always be 1.

Create a new program by selecting:

Functions & Programs > Program Editor > New

Call the program: FactorCount

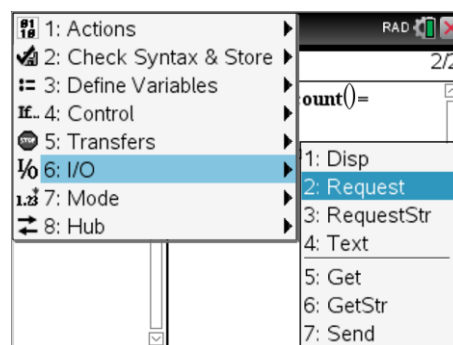
Note that 'FactorCount' is one word as program names cannot contain spaces.



The first task is to request a number from the program user.

Use the I/O (input / output) menu to access the **Request** command. The request command can include a text prompt followed by a variable to store the number.

Request "Enter a number",n



- Quotation marks: " " can be entered by pressing [Ctrl] + [x]
- The comma , can be found in the bottom left corner of the keyboard.

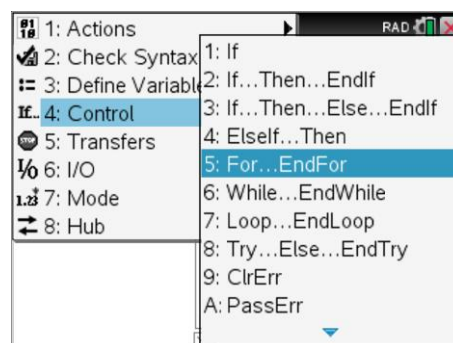
A counter will be used to 'count' the quantity of factors. The counter must be set to zero before the counting process begins.

$c := 0$

Then start a **For** loop by selecting:

Control > For ... EndFor

The loop will start at 1 and finish at n and use i to count the



number of times the loop has been executed.

For $i, 1, n$

An **IF** statement will be used to check if the user's number has a factor each time the program executes the loop.

The IF command can be selected by:

Control > IF ... THEN ... ENDIF

Between the IF and THEN statement insert the command:

Mod(n, i) = 0

Note that 'mod' can be typed directly from the keyboard or accessed through the catalogue.

Move the cursor into the empty line between **THEN** and **EndIf**. This line of code is only executed if **MOD(n, i) = 0** is true.

Insert the command:

c := c + 1

Create another line between **EndFor** and **EndPrgm**

From the I/O menu select **Disp** and type the command:

Disp "Qty Factors", c

Save the program by pressing **[Ctrl] + [B]** and then transfer focus to the calculator application by pressing **[Ctrl] + [Tab]**.

The program is ready to run!

Press the **[VAR]** key and select your program.

Do **NOT** enter the number to be factored into the brackets, press **[Enter]** and enter the number to be factored when prompted by the request command.

Start by testing the number 18 and check the output with the original table.

```

* factorcount 5/7
Define factor count(=
Prgm
Request "Enter a number
c:=0
For i,1,n
If mod(n,i)=0 Then
|
EndIf
EndFor
EndPrgm

```

```

* factorcount 8/8
Define factor count(=
Prgm
Request "Enter a number
c:=0
For i,1,n
If mod(n,i)=0 Then
c:=c+1
EndIf
EndFor
Disp "Qty Factors:",c
EndPrgm

```

```

"factorcount" stored succe
factor count()
Define factor count(=
Request "Enter a number
number
c:=0
For i,1,n
If mod(n,i)=0 Then
c:=c+1
EndIf
EndFor
Disp "Qty Factors:",c
EndPrgm

```

Question: 5.

Determine the quantity of factors for each of the following numbers:

- 24 8 factors ... {1, 2, 3, 4, 6, 8, 12, 24}
- 36 9 factors ... {1, 2, 3, 4, 6, 9, 12, 18, 36}
- 37 2 factors (prime numbers have exactly 2 factors) ... {1, 37}
- 144 15 factors ... {1, 2, 3, 4, 6, 8, 9, 12, 16, 18, 24, 36, 48, 72, 144}

Check each of your answers by writing down all the factors. (All correct, factors listed above)

Question: 6.

Determine the quantity of factors for each of the following numbers. Identify a specific characteristic about the quantity of factors and use this to classify the numbers into two groups, explain your classification.

29 (2), 84 (12), 104 (8), 87 (4), 22 (4), 37 (2), 101 (2), 97 (2), 45 (6), 43 (2), 133 (4), 153 (6), 173 (2), 107 (2).

The numbers: 29, 37, 101, 97, 43 and 173 all have exactly two factors and form the group of 'prime' numbers.

Note that referring to prime numbers as having exactly two factors removes any potential ambiguity with regards to whether or not 1 is prime, it doesn't have exactly two factors; therefore it is not prime. The statements about prime numbers having factors "one and itself" relies on the Boolean logic operator: 'and'. For the Boolean operator AND, True AND True = True, therefore students can interpret factors One (True) AND itself (True) = True.

Question: 7.

Determine the quantity of factors for each of the following numbers. Identify a specific characteristic about the quantity of factors and use this to classify the numbers into two groups, explain your classification.

28 (6), 30 (8), 90 (12), 45 (6), 50 (6), 60 (12), 120 (16), 72 (12), 25 (3), 49 (3), 81 (5), 144 (15), 441 (9), 82 (4), 24 (8), 720 (30)

This classification is harder than the previous one ... students need to pick the 'odd' ones out. If students can see that 25, 49, 81, 144 and 441 all have an odd number of factors they should also identify that all these numbers are perfect squares. Perfect squares are the only numbers that have an odd number of factors since each contains a 'repeated' factor, its square-root.

Question: 8.

The FactorCount program works, but it could be more efficient. Use a stop watch to time how long the program takes to count the number of factors for: 10,000; 20,000 and 30,000. Use these times to predict how long it will take to count the factors for 40,000. Test your answer!

Counting factors for 10,000 takes approximately 14 to 15 seconds.

Counting factors for 20,000 takes approximately 28 to 29 seconds.

Counting factors for 30,000 takes approximately 42 to 43 seconds.

Students should see that each 10,000 numbers is taking just over 14 seconds, averaging close to 14.3 seconds, assuming that the larger numbers do not take any longer, 40,000 should take approximately 57 seconds.

Timed result: \approx 57.02 seconds

Teacher Notes:

The extension to this activity is called Fast Facts. This looks at using a better or improved algorithm to count the number of factors. Adding two improvements to the algorithm reduces the time taken to count the number of factors for 40,000 to a fraction of a second! As a further comparison, counting the factors of 10,000,001 would take the current program a couple of hours. The revised algorithm can count them in less than 5 seconds.