

Program 1: How Many Zeros?

Recall this recent puzzle: If you write out all of the integers from one to one million, how many zeros would you write?

No, this is not a trick question. In other words, the answer is not 0 (because when you *literally* write out “all of the integers from one to one million” as a sentence, there are clearly no zeros). Here's a simpler version: if you write out all of the integers from one to twenty, you will write two zeros: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20.

After having a go at this problem, we discussed a neat way to solve it (that involved a brilliant observation). We also briefly discussed a simple algorithm that would solve this problem. Here it is:

```

n ← 0
count ← 0
repeat
    n ← n + 1
    count ← count + the number of zeros in n
until n is 1 million
display count
    
```

Of course, how can the number of zeros in n be counted? An algorithm for this could be:

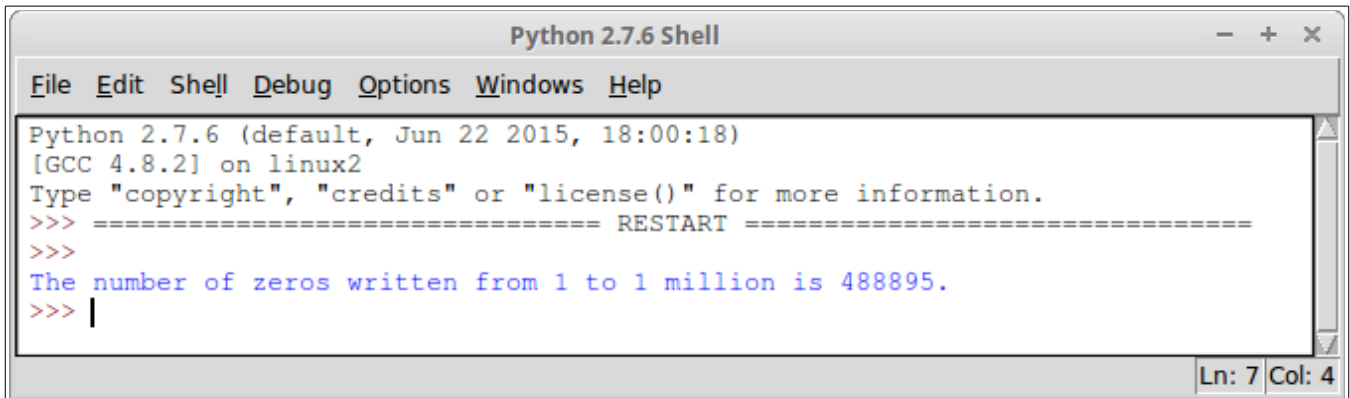
```

zeros ← 0
repeat
    if n % 10 is 0
    then
        zeros ← zeros + 1
    end
    n ← n / 10
until n is 0
    
```

This algorithm checks to see if a remainder exists when n is divided by 10 (i.e., the value of the right-most digit of n). If there is no remainder, then the right-most digit must be 0, and the counter is incremented. The number is then divided by ten (integer division) to remove the right-most digit, and the process continues until n is 0. Take, for example, the number 10,102:

n	remainder	quotient
10,102	2	1,010
1,010	0	101
101	1	10
10	0	1
1	1	0
zeros	2	

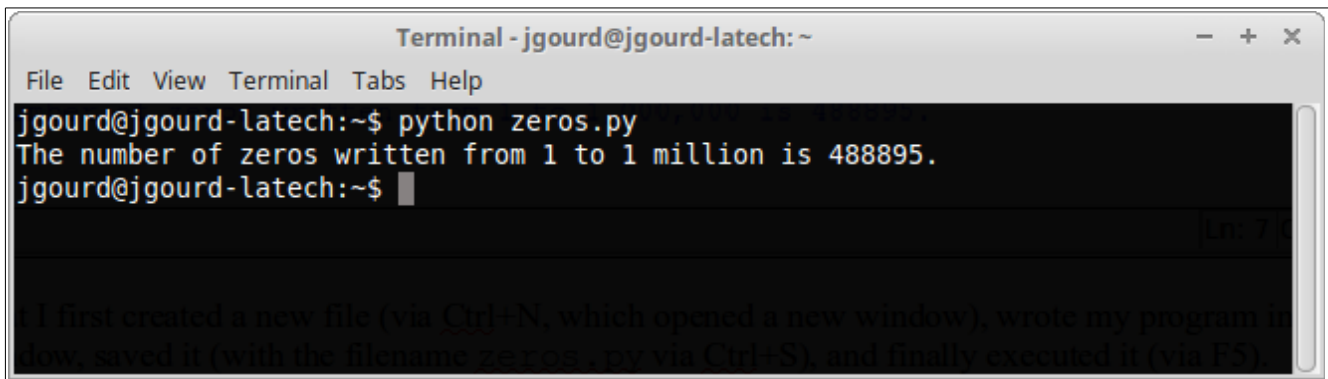
Your task is to write a Python program that implements this (or a similar) algorithm and correctly calculates the number of zeros written from one to one million. Structure your output so that it is like mine. Here is my output in IDLE:



```
Python 2.7.6 Shell
File Edit Shell Debug Options Windows Help
Python 2.7.6 (default, Jun 22 2015, 18:00:18)
[GCC 4.8.2] on linux2
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
The number of zeros written from 1 to 1 million is 488895.
>>> |
```

Note that I first created a new file (via Ctrl+N, which opened a new window), wrote my program in this new window, saved it (with the filename `zeros.py` via Ctrl+S), and finally executed it (via F5).

You can also execute your saved file at the command line (or terminal) as follows:



```
Terminal - jgourd@jgourd-latech: ~
File Edit View Terminal Tabs Help
jgourd@jgourd-latech:~$ python zeros.py
The number of zeros written from 1 to 1 million is 488895.
jgourd@jgourd-latech:~$
```

Note that you must be in the same folder/directory as the file `zeros.py`. In addition, the source file must be run with the Python interpreter. On the RPi (or any machine with the Linux operating system), the Python interpreter can be executed with the command `python`. The full command `python zeros.py` instructs the operating system to execute the Python program defined in the source file `zeros.py` using the Python interpreter.

Homework: How Many Zeros?

Write a Python program that correctly calculates the number of zeros written from one to one million. You may implement the algorithm as described above (or design your own variant). You may choose to implement various functions/methods (or none). The choice is yours.

Make sure to save your Python source code in a file (name it whatever you wish) with a `.py` extension. Also make sure to put an appropriate comment at the top of your program that includes your name, the date that you finished the program, and a short description of what your program does. Don't forget to

appropriately comment your source code as necessary. A template that you can choose to use as a starting point will be provided to you.

Only submit your source code (i.e., the `.py` file).