

Problem Description

This problem is based on the *Collatz conjecture* which is stated as follows:

Begin with any natural number n then follow the sequence

$$n \rightarrow n/2 \text{ (if } n \text{ is even)}$$

$$n \rightarrow 3n + 1 \text{ (if } n \text{ is odd)}$$

indefinitely. The conjecture states that regardless of the initial value of n , the sequence will always eventually terminate with $n = 1$.

So for example, if we start with $n = 13$, we end up with the following sequence:

$$13 \rightarrow 40 \rightarrow 20 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$$

We say that the above sequence is of length 10.

Goals

1. Write a Python script that analyzes the first one million natural numbers (i.e. 1,2,3,...) and finds the integer with the longest sequence.
2. Improve the functionality of your code to print the status of which integer you are checking as the code is running. You likely don't want to print this for every integer you test (that would produce one million lines of printout!) So print it every 1000th integer. Bonus points for figuring out how to erase what has already been printed so that you only get one line of output that is updated with each print command.

Useful Notes

1. There are several fast ways of checking for even or odd. You could try using the `modulo` operator (the `%` symbol in Python) or you could try using some logical operators and some basic knowledge of binary arithmetic. Which is faster?
2. There are many ways of “flushing” the print outs. You could try and use `sys.stdout.flush` and `sys.stdout.write`, or you could open `sys.stdout` using special arguments to `os.fdopen`. Python 3 has special support for this functionality.