

CSE 231, Lab Exercise 3

Partner

Work with a partner on this exercise. Two people should work at one computer. Occasionally switch who is typing.

The Problem

The theme this lab is properties of integers. There is an interesting little algorithm that generates the hailstone sequence (also called the Collatz sequence). It is fairly straightforward. For any positive integer (input):

1. If the number is even, divide it in half.
2. If the number is odd, multiply it by 3 then add 1.
3. If the number is 1, stop; otherwise go to step 1.

The sequence of numbers you generate (the hailstone sequence) always ends with one. Actually, that is not proven, but it is a strong conjecture and has been shown to be true for all numbers less than 3×2^{53} which is pretty good, but not a proof. Note that many sequences have the same sub-sequence embedded in them.

Your Task

Part I: Prompt the user for an integer and display, for that integer, the hailstone sequence. Continue prompting the user for a number to process until a negative number is entered, indicating the user wishes to quit the program. Note that you will need a loop within a loop: one loop where you prompt for the number; the other (inside) loop generates the hailstone sequence.

The first few sequences are:

1	1
2	2, 1
3	3, 10, 5, 16, 8, 4, 2, 1
4	4, 2, 1
5	5, 16, 8, 4, 2, 1
6	6, 3, 10, 5, 16, 8, 4, 2, 1

Here is a sample loop to get you started:

```
number_str = input("Input a number: ")
number = int(number_str)

while number != 1:
    print(number, end=" ")
    number = number - 1

print("End")
```

Show your hailstone solution to your TA to get credit for the lab. The go to Part II.

Part II: Enter two integers that indicate a range to check. Generate the hailstone sequence for each integer in the range and report two values:

- the integer that had the longest hailstone sequence in that range
- the length of the hailstone sequence for that number (not the sequence itself, just the length).