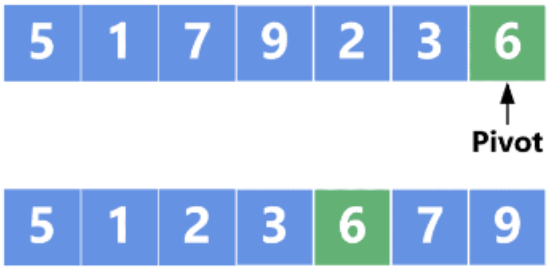
CS211 Lab 9

QUICKSORT



The Collatz sequence for a number n is produced as follows:

if n is even, divide by 2
otherwise multiply by 3 and add 1

The Collatz length of a number is the number of steps it follows before reaching 1. For example, the Collatz length of 9 is 20, because there are 20 terms in the sequence that starts with 9:

9, 28, 14, 7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1

The goal is to sort numbers by their Collatz length. For example, the Collatz lengths of the numbers from 1 to 10 is:

1	2	3	4	5	6	7	8	9	10
1	2	8	3	6	9	17	4	20	7

Which means that when the numbers from 1 to 10 are sorted by their Collatz lengths, they end up in this order (if two numbers have the same Collatz length, the smallest number should come first):

[1, 2, 4, 8, 5, 10, 3, 6, 7, 9]

Imagine **all** the numbers that exist are sorted by their Collatz length.
Write a program that takes in an int *n* and outputs the number in *n*th place.
Find the highest position in this ordering that you can.

Note: **DO NOT** use Arrays.sort – use your own bespoke sorting code, since it's on the exam, you need experience working with it

PEN AND PAPER EXERCISE

Show how the following numbers would be sorted by quicksort, identifying the swaps and pivots involved:

91 60 25 95 69 15 97 41