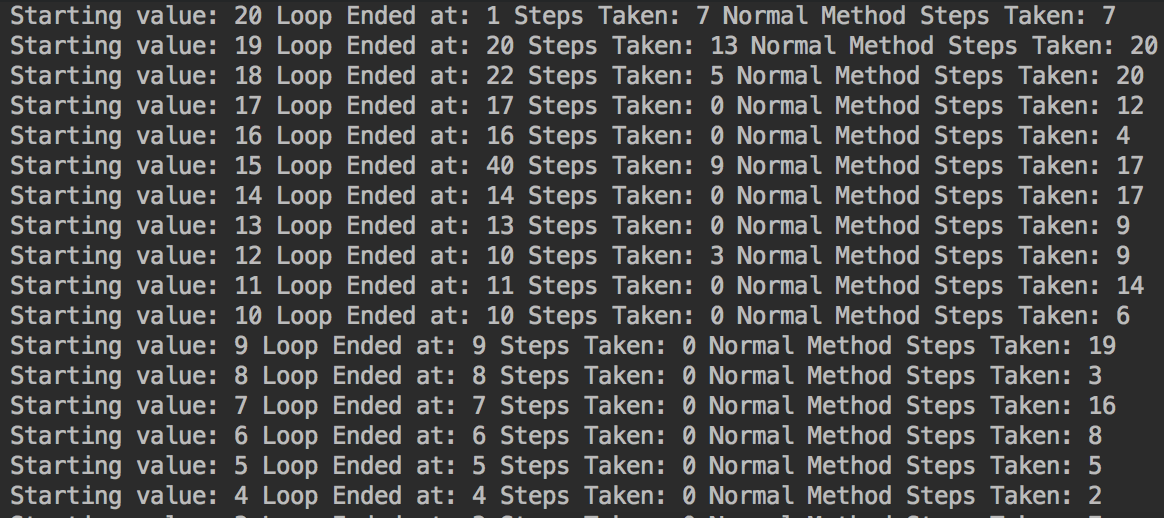
CMPE 365 Lab 2

Robert Saunders

10194030

Please evaluate and run the code, feel free to choose a number to run through the program. The program will run the Collatz algorithm in two ways, one way that runs a recursive method (‘collatz’) to find the number of steps it takes to get to 1, and the other way that utilizes an array to store Collatz values for lookup in subsequent iterations (‘collatzArray’). The output will look similar to below:



As you can see, the output indicates the number of steps taken from both methods. When evaluating which method is better, in terms of performance, it is obvious that the implementation that utilizes an array is much more performant because there are less number of steps for subsequent calculations. This is because the algorithm is utilizing what it already knows from the previous iteration, from this observation we can confidently say that the time complexity is smaller and the average time is faster. We can also say that every terminating run must pass through x = 2 because the algorithm checks between odd and even numbers with a goal of reducing down to one, so the only method to reduce to one from an even number is diving by 2, odd numbers are treated by multiplication so they do not reduce, so x = 2 must be present.