Assignment: Recursion

Part A: Recursion Practice ( 5 pts)

The Collatz Conjecture states that if you take any positive integer, and if that number is even, divide it by 2, and if it is odd, multiply it by 3 and add 1, repeatedly on the results, you will eventually reach 1.

Write a python program that asks the user to enter a positive integer and calls a **recursive** function that returns the number of steps it takes to reach 1.

Part B: 2 algorithms (10 pts)

Write a python function that **Recursively** calculates the nth fibonacci number. The fibonacci sequence is defined as:

f(n > 1 ) = f(n-1) + f(n-2)

f(0,1) = 1

Write a second function that calculates this non recursively.

Using the timer function, calculate and print the runtime it takes to execute your functions to calculate the 30th, 40th, … 100th fibonacci number, and states whether the recursive or non-recursive function is faster for different speeds.

Part C: Dynamic Programming (10 pts).

Write a third function that used dynamic programming to calculate the fibonacci numbers. Use the timer function to compare this to your other approaches.