CSE Template

Rohan Mukherjee

June 26, 2024

4 (c) Recall that the loop condition ensures that curFib is always f_n and prevFib is f_{n-1}. Then in the middle of the loop we store prevFib, namely f_{n-1} in a variable called temp, change prevFib to f_n, and change curFib to $f_n + f_{n-1} = f_{n+1}$. My implementation does the same thing: given that m is still larger than the nth fibonacci number passed into the function as the variable curFib, it calls itself recursively by passing in the new parameters curFib as prevFib and curFib + prevFib as the new curFib. Similarly as above this simply changes prevFib to f_n from f_{n-1} and curFib to f_{n+1} from f_{n-1}. Then when the loop has finished, meaning that m is finally smaller than the nth fibonacci number, we simply return that fibonacci number.

7 (c) We could have added more if/else statements around where we print the messagese to the output (basically just don't define a function and just include the logic around the inner if statement). We could also use string concatenation using the ternary operator to make the string as a constant local variable, and then render that using the ¡p¿ inside the div as before.