Homework 9: Recursive functions

CS16 - Summer 2021

Due:	Thursday, August 26, 2021 (11:59 PM PDT)
Points:	100
Name:	____
Homework buddy:	____

- You may collaborate on this homework with **at most** one person, an optional "homework buddy."
- **Submission instructions:** All questions are to be written (either by hand or typed) *in the provided spaces* and turned in as a single PDF on Gradescope. In other words, you must edit this file directly! Reach out on Slack if you want some suggestions on how to do this. Do not copy and paste the text into a word processor; we will not accept this and your homework may not be graded. If you submit handwritten solutions, write legibly. We reserve the right to give 0 points to answers we cannot read.

1. (3 points) How does a recursive function know when to stop recursing?

2. (4 points) What is a stack overflow? When can it occur? What are the consequences?

3. (3 points) What is a LIFO scheme and how does it relate to stacks?

4. (15 points) Write a definition of a *recursive* function that finds the sum of positive integers that are divisible by 3 in the range (0, `n`] (including `n`). Example, if $n = 12$, then the sum is: $(3 + 6 + 9 + 12) = 30$.

```
int sum3s(int n)
{
```

5. (25 points) Write a definition of a *recursive* function that finds the nth element in the following arithmetic numerical sequence: 3, 11, 27, 59, 123, ... I've started the program for you below, but it is missing the definition. *Hint*: First, figure out the recursive pattern as a linear equation, i.e. $a_n = x * a_{n-1} + y$. You also have to identify the base case. Example outputs would look like this (there is no repeating loop—these are 2 separate runs):

```
Which element of the sequence would you like to know? 4
Element number 4 in the sequence is 59.

Which element of the sequence would you like to know? 7
Element number 7 in the sequence is 507.
```

```
#include <iostream>
using namespace std;
int RecursiveFunc(int num);
int main() {
   int elementN;
   cout << "Which element of the sequence would you like to know? ";
   cin >> elementN;
   cout << "Element number 4 in the sequence is "
        << RecursiveFunc(elementN) << endl;
   return 0;
}

//DEFINITION HERE:
```

6. (25 points) The Fibonacci sequence is defined as a numerical sequence of integers that are the sum of the previous 2 integers. Starting with 0 and 1, the sequence becomes: $0, 1, 1, 2, 3, 5, 8, 13, ...$

   Write the definition of 2 functions: one recursive called `Fibo` (it finds the Nth element in a Fibonacci series) and one non-recursive called `SFS` that calls `Fibo`.

   `SFS` has an integer argument `n`. The pre-condition is that `n` is assumed to be smaller than 256. The post-condition is that SFS prints out all the squares of the Fibonacci sequence of the first `n` elements. For example, calling this line in `main` (just like this): `SFS(7);` will print to standard out: `0 1 1 4 9 25 64`

   (You may assume `<cmath>` is already included.)

7. (25 points) Write a definition of a recursive function that counts the number of the letter ‘z’ or ‘Z’ (i.e. either upper-case or lower-case) in a string. Specifically, given a string variable, `sentence`, when we pass that into a function `CountZ(sentence)`, the function returns an integer that’s a count of the number of the letter ‘z’ or ‘Z’ in the variable `sentence`. This function **must** be a recursive one and cannot contain a loop.