Ain Shams University
Faculty of Engineering
CSE331 Data Structures and Algorithms

Lab 2 report – Dynamic allocation and recursion

1. Objective:

   The student will practice how to

   - Implement a simple algorithm using dynamic arrays (STL vectors)
   - Design and implement recursive functions.
   - Describe the running time of the recursive functions using recurrence relations.

2. Tasks:

2.1. **Tribonacci sequence**:

The Tribonacci sequence Tn is defined as follows:

T0 = 0, T1 = 1, T2 = 1, and $T_n = T_{n-1} + T_{n-2} + T_{n-3}$ for n >= 3.

Given n, return the value of Tn.

Required:

- Write a recursive function to calculate Tn.
- Write the recurrence relation of the Tribonacci function.

2.2. **Fibonacci sequence**:

The Fibonacci numbers, commonly denoted F(n) form a sequence, called the Fibonacci sequence, such that each number is the sum of the two preceding ones, starting from 0 and 1. That is, F(0) = 0, F(1) = 1,

F(n) = F(n - 1) + F(n - 2), for n > 1.

Required:

- Write a function to calculate Fn and consider the range of n.
- What is the time and space complexity for your code?

2.3. **Reverse string recursively**:

you need to design a recursive function that given a string S, returns it in reversed order.

Required:

- Write a recursive function to reverse a string.
- Write the recurrence relation and determine its running time in big O notation.

2.4. **Subarray division**:

Given an array A, find the number of subarrays of length m such that the sum of its elements equals d.

Required:

- Implement your algorithm to solve the problem.
- What's the running time of your algorithm?