Algorithms and data structures

Labwork 4 - Recursion

Dr. Doan Nhat Quang: doan-nhat.quang@usth.edu.vn

Please use Google classroom for labwork submission; join the class by the following link: https://classroom.google.com/c/NTQ1NjMwOTMwNzU0?cjc=gqtgfva

There are 6 Labworks in this course. After each lab work session:

- You need to complete one of the given exercises and upload your files to the assignment "Labwork No. - Group No. - Version 1 (Attendance)". Submission must be done within 30 mins after the lab; otherwise, it will be considered a late submission.

- You will have one week (or 7 days) to complete the remaining exercises and upload your files to the assignment "Labwork No. - Group No. - Version 2 (Complete)"

- Compress all code source files in a zip file and rename it as FULLNAME-ID-Lab#no.zip (e.g NguyenVanA-070-Lab1.zip). Save your files according to the exercise number i.e Ex1.cpp, Ex2.c, etc. Incorrect filenames will result in no score for the respective exercises.

- Only code source files (.c or .cpp) should be in the zip files. Other files (.exe, .o) MUST be removed from the zip file.

- Copy/Paste from any source is not tolerated. The penalty will be applied for a late submission.

NOTE: You must follow the guide. Incorrect zip file names, zip files containing other files (.exe), and copy/pasting lead to heavy penalties.

In this problem, we would like to implement the algorithm to calculate the digit sum of a given natural number that can be used in detecting errors in message transmission or data storage.

For example:

N = 103509, the digit sum = 1 + 0 + 3 + 5 + 0 + 9 = 18.

N = 9512, the digit sum = 9 + 5 + 1 + 2 = 17

Exercise 1 - 6pts:

- Write a pseudo-code to solve the above problem using Iteration.

- Write a program from the pseudo-code and solve the Problem using Iteration.

- Calculate the complexity. Justify your answer.

Exercise 2 - 6pts:

- Write a program to solve the Problem using **Recursion** (with Iteration if necessary).

- Calculate the complexity. Justify your answer.

Exercise 3 - 8pts:
Write a program in C/C++ to enter a natural number n and verify whether n is sphenic using **Recursion**. Calculate the complexity of the proposed algorithm.
Note: A sphenic number is a product of p*q*r where p, q, and r are three distinct prime numbers. Example: $30 = 2 * 3* 5$; $42 = 2*3*7$; $66 = 2*3*11$