DATA STRUCTURES & ALGORITHMS 1

BATCH – A [WEDNESDAY MARCH 13, 2019: 3:30 PM – 6:30 PM]

LAB ASSIGNMENT – 8 CODE:assign08

NOTES:

1. Please carefully read all assignments and there is **no choice**.
2. Use the same files names used in the tutorials, but **add'-xyz'** (assuming 'xyz' is the last three digit of your roll no) as the suffix of the filenames.
3. **Follow variable and function naming conventions**
    a. except for variables in for-loop, none of the other variables should be a single character.
    b. The variable names and function names should indicate what they are storing/computing. For this assignment, we have given you some of the variable names and function names to use. They are highlighted as function_name or variable_name
    c. All global variable should start with 'g_'
4. Indentation improves readability. Please pick an indentation style and **indent your code** appropriately.
5. Follow constants and type naming
    a. All constants should be defined using IFNDEF and DEFINE
    b. All structures should have a TYPEDEF to a simpler name
6. When in doubt about naming or style conventions, consult the following link: https://users.ece.cmu.edu/~eno/coding/CCodingStandard.html

---

PROBLEMS [Total Marks: 20]:

Today we are going to practice how to write modular C code. So, far we have created one C file which contains: global variables and function declarations, type and constant declarations and a main function. The primary objective of today is to know how to split interfaces and implementations into separate files and to manage all the source code as a single project, with the help of make files.

Exercises:

1. *[6 Marks]* Familiarize yourself with the simple makefile tutorial in the link:http://www.cs.colby.edu/maxwell/courses/tutorials/maketutor
    a. Follow along the tutorial to create a project with the code provided in the tutorial
    b. Experiment with different versions of make files

(Warning: You will be asked to explain the meaning and purpose of the different flags and macros)

2. *[7 Marks]* Familiarize yourself with the modules and compilation tutorial in the link: https://www.cs.bu.edu/teaching/c/separate-compilation/
    a. Follow along the tutorial to create a project with the code provided in the tutorial
    b. Pay special attention to problems highlighted in the tutorial including multiple inclusion problem (Now you can appreciate IFNDEF)
3. *[7 Marks]* Implement the following functions to improve “the grades” project
    a. Declare and implement the following
        i. Min grade
        ii. Max grade
        iii. Sum of the grades (two variants)
            1. Iterative
            2. Recursive
        iv. Average grade(use the sum function you wrote)
    b. Improve the makefile as necessary

Note: penalties for violating style and naming conventions are doubled for today