PROJECT DESCRIPTION

Deadline: Tuesday, December 17, 2019 by 2359hrs **Total Marks=20**

General Instructions

This is a group project. Please make groups of max 3 students each. You are strongly advised to make balanced groups, with an objective to facilitate/provide peer-help to students with relatively weaker programming skills. Please submit the consolidated list to your Course TA by Friday Morning, November 29th, 2019.

There are following 3 *options*. Each group will be randomly assigned to a project once you submit your groups.

PROJECT OPTION 1

Write a C++ program which creates a quadrilateral. In case it is a rectangle, it is defined by the floating-point attribute's **length** and **width**, each of which has a default value of **1,** with a minimum value of **1** and a maximum value of **10.0.** The quadrilateral can also be defined by the **Cartesian coordinates** of the four corners of the rectangle.

The program implements the following functions, each of which **MUST** be called by a menu (implemented using C++ Case statement):

1- A function which prompts the user to enter the quadrilateral using either format. [2 marks]

In case of rectangle

2- A function which calculates and returns the **perimeter** of the rectangle. [2 marks]
3- A function which determines if the rectangle is a square or not. [2 marks]
4- A function which **draws** the outline of the Rectangle. This can be done by printing the perimeter of the rectangle using some special characters. [4 marks]
5- A function which **fills** the Rectangle. This can be done by filling the rectangle by special characters (which are different from the characters chosen for the perimeter). [4 marks]

In case of a non-rectangular quadrilateral

6- A function which accepts 4 sets of **Cartesian coordinates** and returns the **length of the maximal diagonal** across. [3 marks]
7- A function which computes and prints the perimeter of the quadrilateral. [3 marks]

PROJECT OPTION 2

A library is a place where a user, once registered, can loan (and return) up to a certain number of books. For a user to loan a book, he/she must be registered, and a copy of the book must be present with the library. The library can add/delete a user, search for a book, search to verify how many books have been issued to a user, loan a book and receive a book back from a user. Assume a maximum of 10 books and 10 users can exist in the library at any one time (i.e. limit the number of books and users to 10). Also, assume a user can loan at max 3 books.

You're required to write a C++ code functions that will perform the following tasks.

1. Add a new user to the system [2 marks]
2. Delete an existing user from the system [2 marks]
3. Search a user and display the number of outstanding books against him/her. [3 marks]
4. Loan a book to a user (both book and user must exist). [3 marks]
5. Return a book from a user. [3 marks]
6. Display a list of existing users (in sorted order) [4 marks]
7. Display a list of existing books and their status (available or borrowed). [3 marks]

   The program MUST provide a friendly user interface to perform the above tasks. You can hardcode a list of 10 books in the system or read from a file.

PROJECT OPTION 3

Write a C++ to program to make a basic calculator.

You're required to write a C++ code functions that will perform the following tasks.

1. Insert 2 numbers from users. [1 marks]
2. Add, Subtract, divide and multiply inserted numbers. [2 marks]
3. Calculate n^{th} power and square root of a number, i.e., (X^N) & ($\sqrt{X}$) [4 marks]
4. Find the factorial of a number. (must use recursion) [3 marks]

Perform the following tasks after reading an array from user (size of array =10)

5. Sort the array. [3 marks]
6. Find mean, mode and median of array. [3 marks]
7. Should keep the history of previous 3 calculation (operation and result) [4 marks].

The program MUST provide a friendly user interface to perform the above tasks

Submission Instructions

1- Well documented single C++ source files. (Both the hard copy as well as error-free ready-to-run soft copies). The program **MUST** have a reasonably good menu-driven user-friendly interface.
2- An executable file of the program output.
3- After the project submission, there will be a viva/presentation/quiz based on what you have learned from the project. The format and marks for this part will be decided later.

Upload all the files (zipped together) as a single file to the link provided on Moodle as well as submit hard copies of the C++ source files to your TA, before the deadline expires. The link will automatically close after the deadline and no late submissions will be allowed.

Best of luck!!!