COSC 2437 - Data Structures

Homework Assignment 1

Total points = 100

Classes *Review*

Make sure that you follow a good programming style. In each *main* program, include a header comment with the following information:

```
//*********************************************************************
//Name:
//Class: COSC 2437 F20
//Instructor: Marwa Hassan
//Assignment x
//Date:
//Program description: WRITE what this program actually does
//**********************************************************************
```

Include a header comment for header files and classes. In addition to proper function documentation.

For review material, refer to Lecture Notes → 1436 Lecture Notes - Classes

Problem 1 (25 points):

File: circle.cpp (class declaration, implementation and main in one .cpp file)

Write Circle class, with proper instance variables and member functions. In main function, allocate an array of Circle objects. Get the size of the array from the user. Also, initialize each circle from the user. Find the largest circle in the array based on the area.

Problem 2 (35 points):

Files: LocationRevenue.cpp - LocationRevenue.h – restaurant.cpp

Consider a restaurant with four locations. Create `LocationRevenue` class that keeps income data for a specific location. The class contains the following members:

- a member array for holding quarterly income figures.
- a *static* member for maintaining the restaurant's gross income - *other design options are fine as well.*
- a member function for setting the four quarterly values. *Make sure that the gross income is updated correctly.*
- a function that takes an integer argument and returns the corresponding quarterly income of the location.

In main function (restaurant.cpp), create an array of `LocationRevenue` objects. Read the income figures for each location from the user. Display each location's sale and the restaurant's sale for the year.

Validate entered data.

Problem 3 (40 points):

```
Files: - Measure.cpp & Measure.h (or just Measure.h)
       - mainMeasure.cpp
```

Write and demonstrate the class `Measure`. It has three instance variables: yards, feet, and inches. *Note: 1 foot = 12 inches. 1 yard = 3 feet.*

Include the following member functions *(refer to lecture notes, FeetInches class)*

- default constructor - initializes the instance variables to zeros
- non-default constructor – takes the 3 arguments
- setter and getter functions

Overload the following operators in `Measure` class:

1. + add two `Measure` objects and return the sum
2. + add a `Measure` object and an integer value representing inches and return the sum
3. - subtract the argument object from the invoking object and return the difference
4. = the assignment operator
5. compare two `Measure` objects by overloading the operators:
   - ==
   - <
   - >
   - !=
6. += add a `Measure` object argument to the calling object (e.g.: `obj1 +=obj2;`)
7. += add an integer value representing the inches to the calling object (e.g.: `obj1 +=8;`)
8. The stream operators: << & >>

What to submit?

Place the files of homework assignment in the folder firstInitialLastNameAssignX, where X is the homework number. Example, for the name Joe Smith, the folder name must be **jSmithAssignX**.

Refer to the document "Separating Specification from Implementation" under Course Content.

The submitted **zipped** folder should contain:

Problem 1 **file:**

- circle.cpp

Problem 2 folder contains:

- *LocationRevenue.cpp*
- *LocationRevenue.h*

- *restaurant.cpp*
- makefile [-10 points if not included]

Problem 3 folder contains:

- Measure.*cpp & *Measure.*h (or just Measure.h)*
- *mainMeasure.cpp*
- makefile [-10 points if not included]

Grading criteria for submitted code

	Grading
Program correctness and code requirements	70
Programming Style	15
User Interface	5
Documentation	10
Total	100

*Two points will be deducted from the total assignment grade for every instruction that is not followed.

Late work penalty: 25% if one day late (up to 24 hours late); 50% if two days late (from 24 to 48 hours late); zero credit if more than two days.