**The objective of this lab is to:**

1. Understand and practice composition and aggregation.
2. Practice good coding conventions e.g commenting, meaningful variable and functions names, properly indented and modular code.

**Instructions!**

1. This is a **graded** lab, you are strictly **NOT** allowed to discuss your solutions with your fellow colleagues, even not allowed asking how is he/she is doing, it may result in negative marking. You can **ONLY** discuss with your TAs or with me.
3. Strictly follow good coding conventions (commenting, meaningful variable and functions names, properly indented and modular code.
4. Save your work frequently. Make a habit of pressing **CTRL+S** after every line of code you write.

## Task 01: [15 Marks]

***Note:*** *Create multi-file project for this task i.e. the declaration and implementation of each of the following classes should be in separate .h and .cpp files.*

Create a class named ***Teacher***. It should contain the name of the teacher and the teacher's office extension number (an integer) as member variables. Implement constructor and other required functions for this class.

Then, create a class named ***Classroom***. It should contain a room number (an integer) and the capacity (an integer) as member variables. Implement constructor and other required functions for this class as well.

Finally, create a class named ***Course***. A Course contains a course title, course code, a Teacher, and a Classroom. Its constructor requires the course title, course code, the name and office extension number of the Teacher, and the room number and capacity of the Classroom. Write a function named *changeTeacher* to change the teacher for a particular course. Write a function named *allocateClassroom* to allocate a new room to a course.

Each of the above three classes should have a member function which should display all attributes of a given object. Also, implement the copy constructor and overload assignment operator for each of the above three classes. Demonstrate your work in *main( )*.

## Task 02: [15 Marks]

The Westfield Carpet Company has asked you to write an application that calculates the price of carpeting for rectangular rooms. To calculate the price, you multiply the area of the floor (width times length) by the price per square foot of carpet. For example, the area of floor that is 12 feet long and 10 feet wide is 120 square feet. To cover that floor with carpet that costs $8 per square foot would cost $960. (12 × 10 ×8 = 960.)

First, you should create a class named **RoomDimension** that has two **FeetInches** (this class is given below) objects as attributes: one for the length of the room and one for the width. (you can overload multiply or other operators in FeetInches class if required). The RoomDimension class should have a member function that returns the area of the room as a FeetInches object.

Next, you should create a **RoomCarpet** class that has a RoomDimension object as an attribute. It should also have an attribute for the cost of the carpet per square foot. The RoomCarpet class should have a member function that returns the total cost of the carpet.

Once you have written these classes, write code main that asks the user to enter the dimensions of a room and the price per square foot of the desired carpeting. Then it should display the total cost of the carpet.

```
// class FeetInches.

class FeetInches
{
       int feet;
       int inches;

public:
       FeetInches(int f=0, int i=0)
       {
           feet = f;
           inches = i;
       }
       void setFeet(int val)
       {
            feet = val;
       }
       void setInches(int val)
       {
            inches = val;
       }
       int getFeet()
       {
            return feet;
       }
       int getInches()
       {
            return inches;
       }
};
```

```
// class RoomDimension.

class RoomDimension
{
       FeetInches length;
       FeetInches Width;

public:
       // Constructor and other
       supported functions.

       FeetInches calculateArea();

};


// class RoomCarpet.

class RoomCarpet
{
       RoomDimension length;
       float cost_per_square_foot;

public:
       // Constructor and other
       supported functions.

       float calculateTotalCost();

};
```

*Shoot for the moon. Even if you miss, you'll land among the stars*
*-Les Brown*