CMP 142 – Object Oriented Programming
Fall 2019
LAB-04

Issue Date: October 12, 2020

The objective of this lab is to:

1. Practice and understand simple classes with constructors and destructors.
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: [10 Marks]

Create a class POINT with two private integer data members named x and y. Implement following public member functions for this class:

- Setters/mutators for each data member.
- Getter/acessors for each data member.
- A default constructor (with no arguments) that should initialize both data members to 0.
- Parameterized constructor having 2 arguments.
- An *isValid* function that should return true if point is a valid point and return false if it is not. A point is a valid point if both x and y are positive values.
- A *printStars* function that should display x times stars in x direction, and y times stars in y direction only if the point is a valid point i.e both of its data members hold positive values.
- A main function to demonstrate the functionality of Point class.

Task 02: [15 Marks]

Implement the following class to store data of a student.

```
class PUCITstudent
{
private:
    string name;         // to store name of the student
    string rollNo;       // to store roll number of the student
    float cgpa;          // to store cgpa of the student

public:
    // A constructor that should act as default as well as parametrized constructor.
    void setName();      // set name of the student
    void setRollNo();    // set roll number. It should be of exactly 10 characters
    void setCGPA();      // set CGPA. It should be strictly between 1-4.00
    string getName();    // return name
    string getRollNo();  // return roll number
    float getCGPA();     // return CGPA
    string getDegreeProgram(); // return degree program in which student is enrolled
    string getBatch();         // return batch of the student
    char getSection();         // return section (M for morning and A for Afternoon)
    void displayStudentStatus();     // student status i.e pass, drop out, probation
    int graduatingYear();      // return year of graduation
};
```

Roll Number should be of exactly 10 characters and of the form BITF14M524. Where first three letters of the roll number represents degree program in which the student is enrolled. PUCIT is currently offering 4 degree programs at undergrad level i.e BIT, BCS, BSE, MIT. Next three letters represents batch. BIT, BCS and BSE are 4 years degree programs while MIT is 2 years degree program. So you can compute graduating year by manipulating batch and degree program. Section can also be determined by parsing roll number.

Status of the student can be determined by his/her CGPA. If CGPA is greater than 1.7 then student is considered as pass. If it is less than 1.5 then drop out and if it is between 1.5 and 1.7 then student is considered on probation.

Sample Execution 1:

```
Enter name: Umair Farooq
Enter roll No: BITF12M523.
Enter CGPA: 3.64

Name:                Umair Farooq
Roll Number:         BITF12M523
Degree Program:      BIT
Batch:               FALL 2012
Section:             Morning
Status:              Pass
CGPA:                3.64
Year of graduation:  2016
```

Sample Execution 2:

```
Enter name: Ali Hassan
Enter roll No: MITF10A552.
Enter CGPA: 1.32

Name:                Ali Hassan
Roll Number:         MITF10A552
Degree Program:      MIT
Batch:               FALL 2010
Section:             Afternoon
Status:              Drop out
CGPA:                1.32
Year of graduation:  2012
```

Note: If you are not aware of string manipulation functions you can take help from internet.

Task 03: [15 Marks]

Write a class ACCOUNT that represents your bank account and then use it. The class should allow you to deposit money, withdraw money and send you an error message if you go overdrawn etc. Add a function to show your account details including your available balance. Also, add constructors and destructors to your class.

You may create following data members for your class:

```
int accountNo;          // Account number (taken from user)
```

```
string accountTitle;     // Name of the account holder (taken from user)
double openingBal;       // Minimum Bal should be Rs.5000. (do not take from user)
string emailID;          // Optional attribute. If user do not provide assign
                               "Email ID Not Available"
double accountBal;       //The accountBal should never go below the openinBal
```

Do not write setters for openingBal and AccountBal. AccountBal can only be updated by withdraw or deposit functions.