

Week #7

Objective: For students to get some practice of:

- Problem solving with classes
 - Writing formal Class description
 - .1 Determining data types for member data
 - .2 Member Function prototypes
 - Translating class descriptions into class declarations
 - .1 Use of private, public keywords.
 - .2 Data/information hiding
 - Working with objects in various applications
 - .1 Calling member functions
 - .2 Run-time/actual arguments
 - .3 Default-arguments
- Inheritance:

Theory:

A class can be viewed as a customized 'struct' that **encapsulates data** and **function**. Format of a class definition:

```
class your_class_name
{
    member_access_specifier:
        data members;
    member_access_specifier:
        member_functions();
};
```

When cin and cout are used to perform I/O, actually objects are created from istream and ostream respectively that has been defined in iostream header file.

Building a Class

```
#include <iostream>
using namespace std;
class smallobj                                //declare a class
{
private:
    int somedata;                               //class data
public:
    void setdata(int d)                         //member function to set data
    { somedata = d; }
    void showdata()                             //member function to display data
    {
        cout << "Data is " << somedata << endl; }
};

void main()
{
    smallobj s1, s2;                            //define two objects of class smallobj
    s1.setdata(1066);                          //call member function to set data
    s2.setdata(1776);
    s1.showdata();                             //call member function to display data
    s2.showdata();
}
```

}

Lab Exercise:

Q #1: You are already familiar with the `Rectangle` and `DormRoom` object types and have also written class descriptions for the two classes. Write down C++ class declarations for both the class.

Q #2: A company called Restaurant Products, Inc. wants to produce small machines for use by waiters in a restaurant. Each machine is initialized with the name of the restaurant and the local sales tax rate. When a customer has finished eating, the waiter enters the cost of the meal and has the machine output a bill in the following form:

```
Bill
Meal cost      xx.xx
Tax            xx.xx
Tip            xx.xx
Total cost     xx.xx
```

The bill automatically adds a 15% tip to the cost. After this the waiter enters the amount paid by the customer and outputs a receipt that includes the following information in the following form:

```
Receipt
Total cost     xx.xx
Payment        xx.xx
Change         xx.xx
```

```
Thank you for dining at <restaurant name>.
```

To produce the software for this machine write down the class description and declaration for the `MealBill` class.

Q #3: Write a program for The University Summit Restaurant, which allows a student Ahmed Ali to charge the cost of a meal to his student account. Records are stored in the student account object `ahmedAli` that has a current balance of Rs.2000.00. The Summit Restaurant implements billing by creating a `MealBill` object, which is set to charge 6% sales tax.

The program first prompts for the cost of the meal and then computes and prints the bill. It then makes a charge to the student's account and prints the receipt. In the end the program prints the current status of Ahmed's account.

Sample run:

```
Amount of Ahmed's bill: 85.5
```

```
Bill
Meal cost      65.5
Tax            xx.xx
Tip            xx.xx
Total cost     xx.xx
```

Charge the meal to Ahmed's account

Receipt

Total cost	xx.xx
Payment	xx.xx
Change	xx.xx

Thank you for dining at University Summit

Current status of Ahmed's account:

ID:	0210-BCS-06
Name:	Ahmed Ali
Balance:	Rs.xx.xx

Q #4: A student's grade record is maintained by the registrar in the Academic Dept. The record includes the studentID along with the total number of credits attempted and the total grade points earned by the student. The grades points are determined by the scale A(4), B(3), C(2), D(1), and F(0).

E.g. 4-credit course with grade B: $\text{grade points} = 4 * 3 = 12$

A student's grade record is initialized with his/her ID, number of credits and the grade points earned. The number of credits and grade points default to 0 (GPA 0.0), for a new student. The student record is used for the purpose of

- (i) determining the GPA – the total credits and the grade points are used to compute the GPA.
- (ii) to write the grade point information in the following format:

Student: 0210-BCS-01 Units: 100 GradePts: 345 GPA: 3.45

Each semester the registrar updates the student records with grades from recently completed courses. For instance, assume student "0210-BCS-01" completed a semester in which he/she earned 16 credits and 58 grade points. So, credits and gradePts are now increased to 116 and 403, respectively.

Write down the description and declaration for the GradeRecord class.

Q #5: Write a program that creates a student grade record for a new student. To indicate that student is new print the student's initial grade point average. The program then prompts for the grade points and credits that the student has earned in the first semester and updates his/her grade record. The student's record at the end of the semester is then output to the screen.

Sample Run:

Ali's GPA is 0.00
Ali's 1st semester grade points and units: 18 55

Student: 0210-BCS-01 Units: 18 GradePts: 55 GPA: 3.06

Q.6

Create a Base Class Person having attributes(name,age,gender) with behavior of showdata(displaying all the members) and override same method in derived classes.

Derive student Class and from student class derive graduate student with appropriate data members and behavior.