CS 201: Problem Solving & Programming II

Lab #4

**Exercise 1: Class Inheritance and Composition**

Suppose our CS-201 course is being maintained by a system, which holds students, instructor and course information. The System can be depicted in this way:

**Person**

*Name, DOB*

**Student**

*Id, CGPA*

**Instructor**

*Office, Designation*

**Course**

*Id, Name, StudentList, Instructor*

The following design constraints are to be followed:

1. The Person class has two attributes: name and Date of birth (DOB).
2. The Student class inherits from Person class and has following attributes: ID, CGPA.
3. The Instructor class also inherits from Person class and has following attributes: Office, Designation.
4. The Course class has following attributes: Id, Name, Students and Instructor. The S*tudentList* and I*nstructor* attributes are two reference of *Student* and *Instructor* class respectively. The S*tudentList* is an array of *Student* objects.

We can populate the objects of these classes using data from the following files:

The “course.txt” file which will contain data in the following format:

Course name

ID

Number of students

The “students.txt” file will contain the list of students enrolled in that course in this format:

Student Name

ID

DOB

CGPA

There won’t be more than 20 students in the course.

The “instructor.txt” file will contain the information of the instructor of that course:

Instructor name

DOB

Office

Designation

**Putting array of objects within an Object:**

For this exercise we may have to put an array of objects inside another Object. To demonstrate how to achieve that we may consider the following: let’s say, we have a Basket object which holds Apple objects in it. We can define the code of Basket and Apple in this way:

Apple {

Public:

int color;

double size;

};

Basket {

Private:

Apple apples[20];

int numOfApples;

Public:

void setApples(Apple appleObjects[], int numOfApples);

};

void Basket::setApples(Apple appleObjects[], int numOfApples){

for(int i = 0; i<numOfApples; i++) {

apples[i] = appleObjects[i];

}

}

**Converting string to double:**

The following *atof()* function can convert a decimal value stored in string variable to a double variable.

string cgpaStr = “3.99”;

double cgpa = atof(cgpaStr.c\_str());

The c\_str() method returns a pointer of the string variable.