Experiment 1:

a)Analyse and design classes for a student management system.

```
#include<iostream>
#include<string>
using namespace std;
class student{          //student class
  private:
  string id,branch,sem;
  public:
  student(string id,string branch, string sem){

  }
};
                        //course class
class course{
  pivate:
  string c_name,sem;
  int c_id;

public:
    course(string c_name,string sem, int c_id){

    }
};

class faculty{
  private:
  string id,branch,email;
  public:
  faculty(string id ,strung branch,string email){

  }
};
```

b) To create a student class, its constructor, destructor and member functions

```
#include<iostream>
using namespace std;
class student
{
    int rollno;
    char name[50];
    int eng_marks, math_marks, sci_marks, lang2_marks, cs_marks;
    double average;
    char grade;
     public:
     student ();                              // Creation of Constructor
    ~student();                              //Destructor
    void getdata();                     // Declaration of Member function to insert data
    void showdata() const;       // Declaration of Member function to display data
    void calculate();                 // Declaration of Member function for calulating
marks
    int retrollno() const;
};
student :: student()                 //constructor
{
        cout<<"\nThis is Student Details constructor called..........."<<endl;
}
void student::calculate()
{
    average=(eng_marks+math_marks+sci_marks+lang2_marks+cs_marks)/5.0;
    if(average>=90)
    grade='A';
    else if(average>=75)
    grade='B';
    else if(average>=50)
    grade='C';
    else
    grade='F';
}

void student::getdata()
{
    cout<<"Enter student's roll number: ";
    cin>>rollno;
    cout<<"Enter student's name: ";
    cin.ignore();
    cin.getline(name,50);
    cout<<"\n\nEnter Student's Marks"<<endl;
    cout<<"All marks should be out of 100";
    cout<<"\n\nEnter marks in English: ";
    cin>>eng_marks;
    cout<<"Enter marks in Math: ";
    cin>>math_marks;
    cout<<"Enter marks in Science: ";
    cin>>sci_marks;
    cout<<"Enter marks in Second language: ";
    cin>>lang2_marks;
```

```
    cout<<"Enter marks in Computer science: ";
    cin>>cs_marks;
    calculate();
}

void student::showdata() const                  //function definition for displaying the
data
{
    cout<<"\nRoll number of student : "<<rollno;
    cout<<"\nName of student : "<<name;
    cout<<"\nEnglish : "<<eng_marks;
    cout<<"\nMaths : "<<math_marks;
    cout<<"\nScience : "<<sci_marks;
    cout<<"\nSecond Language : "<<lang2_marks;
    cout<<"\nComputer Science :"<<cs_marks;
    cout<<"\nAverage Marks :"<<average;
    cout<<"\nGrade of student is :"<<grade;
}

int student::retrollno() const
{
return rollno;
}

void create_student();
void display_sp(int);//display particular record
void display_all(); // display all records
void delete_student(int);//delete particular record
void change_student(int);//edit particular record
student :: ~student()
{
        cout<<"\n\nStudent Detail is Closed.............\n";
}
int main()
{
    student s;
    s.getdata ();
    s.showdata ();
    s.calculate ();
    return 0;
}
```

c)To create a course class, its constructor, destructor and member functions

```
#include<iostream>
using namespace std;
class course
{
int credits;
char course_name[50];
char course_code[50];
     public:
     course ();                                          //Constructor
    ~course();                                           //Destructor
    void input();
    void display() const;

   };
course :: course()
{
        cout<<"\nThis is Course Details constructor called.........."<<endl;
}
void course::input()
{
    cout<<"Enter the Course Code: ";
    cin>>course_code;
    cout<<"Enter Course name: ";
    cin.ignore();
    cin.getline(course_name,50);
    cout<<"Enter credits:";
    cin>>credits;
}
void course::display() const
{
    cout<<"\nCourse Code: "<<course_code;
    cout<<"\nCourse Name : "<<course_name;
    cout<<"\nCredits : "<<credits;
}
course :: ~course()
{
        cout<<"\n\nStudent Detail is Closed.............\n";
}
int main()
{
    course c;
    c.input ();
    c.display();
   return 0;
}
```