#include <iostream>

#include <fstream>

#include <cstring>

#include <iomanip>

#include<cstdlib>

#define max 50

using namespace std;

class Student

{

    char name[max];

    int rollNo;

    int year;

    int division;

    char address[50];

    friend class FileOperations;

    public: Student()

            {

                strcpy(name,"");

                rollNo=year=division=0;

                strcpy(address,"");

            }

            Student(char name[max],int rollNo,int year,int division,char address[max])

            {

                strcpy(this->address,address);

                strcpy(this->name,name);

                this->division=division;

                this->rollNo=rollNo;

                this->year=year;

            }

            int getRollNo()

            {

                return rollNo;

            }

            void displayStudentData()

            {

                cout<<endl<<setw(3)<<rollNo<<setw(10)<<name<<setw(3)<<year<<setw(2)<<division<<setw(10)<<address;

            }

};

class FileOperations

{

    fstream file;

    public:FileOperations(char \*name)

           {

             //strcpy(this->name,name);

             this->file.open(name,ios::in|ios::out|ios::ate|ios::binary);

           }

           void insertRecord(int rollNo,char name[max],int year,int division,char address[max])

           {

               Student s=Student(name,rollNo,year,division,address);

               file.seekp(0,ios::end);

               file.write((char\*)&s,sizeof(Student));

               file.clear();

           }

           void displayAllRecords()

           {

               Student s;

               file.seekg(0,ios::beg);

               while(file.read((char \*)&s,sizeof(Student)))

               {

                   s.displayStudentData();

               }

               file.clear();

           }

           void displayRecord(int rollNo)

           {

               Student s;

               file.seekg(0,ios::beg);

               void \*p;

               while(file.read((char \*)&s,sizeof(Student)))

               {

                   if(s.rollNo==rollNo)

                   {

                       s.displayStudentData();

                       break;

                   }

               }

               if(p==NULL)

                   throw "Element not present";

               file.clear();

           }

           void deleteRecord(int rollNo)

           {

               ofstream newFile("new.txt",ios::binary);

               file.seekg(0,ios::beg);

               bool flag=false;

               Student s;

               while(file.read((char \*)&s,sizeof(s)))

               {

                   if(s.rollNo==rollNo)

                   {

                       flag=true;

                       continue;

                   }

                   newFile.write((char \*)&s,sizeof(s));

               }

               if(!flag)

               {

                   cout<<"Element Not Present";

               }

               file.close();

               newFile.close();

               remove("student.txt");

               rename("new.txt","student.txt");

               file.open("student.txt",ios::in|ios::ate|ios::out|ios::binary);

           }

           ~FileOperations()

           {

               file.close();

               cout<<"Closing file..";

           }

};

int  main()

{

    ofstream newFile("student.txt",ios::app|ios::binary);

    newFile.close();

    FileOperations file((char \*)"student.txt");

    int rollNo,year,division,choice=0;

    char name[max],address[max];

    while(choice!=5)

    {

        //clrscr();

        cout<<"\n\*\*\*\*\*Phone Book\*\*\*\*\*\n";

        cout<<"1) Add New Record\n";

        cout<<"2) Display All Records\n";

        cout<<"3) Display by RollNo\n";

        cout<<"4) Deleting a Record\n";

        cout<<"5) Exit\n";

        cout<<"Choose your choice : ";

        cin>>choice;

        switch(choice)

        {

            case 1 : //New Record

                    cout<<endl<<"Enter RollNo and name : \n";

                    cin>>rollNo>>name;

                    cout<<"Enter Year and Division : \n";

                    cin>>year>>division;

                    cout<<"Enter address : \n";

                    cin>>address;

                    file.insertRecord(rollNo,name,year,division,address);

                    break;

            case 2 :

                    file.displayAllRecords();

                    break;

            case 3 :

                    cout<<"Enter Roll Number";

                    cin>>rollNo;

                    try

                    {

                        file.displayRecord(rollNo);

                    }

                    catch(const char \*str)

                    {

                        cout<<str;

                    }

                    break;

            case 4:

                    cout<<"Enter rollNo";

                    cin>>rollNo;

                    file.deleteRecord(rollNo);

                    break;

           case 5 :break;

        }

    }

}