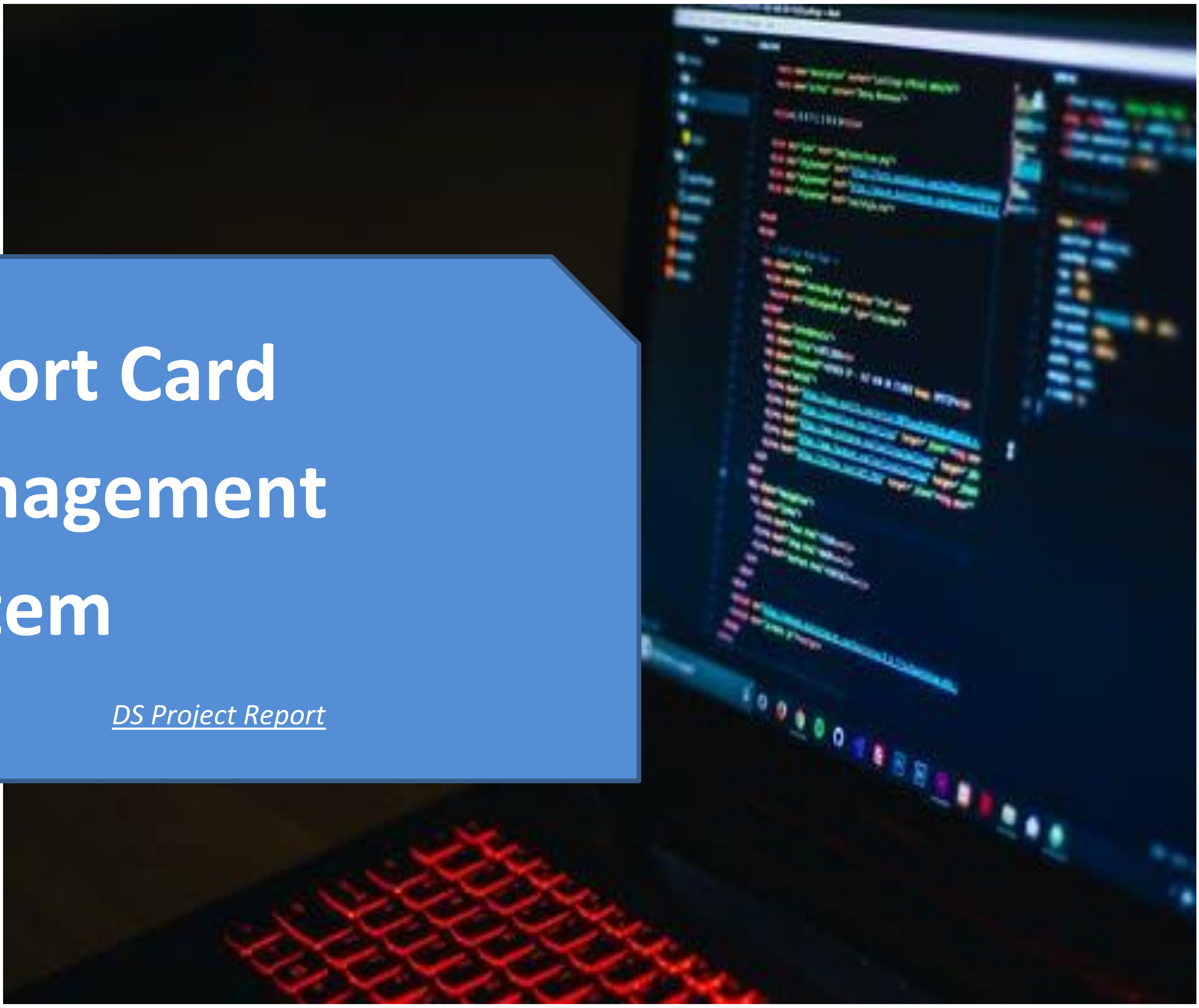


Report Card Management System

DS Project Report

PROPOSED TO:

INDU SINGH MAM

PREPARED BY:

LAKSHAY

2K19/SE/067
KUSHAL JAIN
2K19/SE/066

CONTENTS

TOPIC PAGE NO.

- ACKNOWLEDGEMENT 1
- INTRODUCTION 2
- PROJECT OBJECTIVES 3
- SOFTWARE AND HARDWARE REQUIREMENTS 4
- THEORY OF CONCEPTS USED 5- 9
- SOURCE CODE 10-20
- OUTPUT SCREENS 21-26
- ADVANTAGES AND DISADVANTAGES 27
- REFERENCES 28

ACKNOWLEDGEMENT

The way can't walk itself. We have to walk on it. For that we must have a guide. Many guides have contributed to the successful completion of the project we would like to place on record my grateful thanks to each one of them who help us in this project.

Before we get into thick of the thing, we would like to add a few heartfelt words for the people who gave us unending time support whichever and whenever necessary, our grateful thanks go to our dept. ,which provides us an opportunity as a project subject in 3" semester to develop a report work skill in this system analysing .

We would like to thank our parents & friends for giving us full feedback when we are in trouble.

Our special thanks go to Indu Singh Mam to give their expert guidance to us whenever necessary.

INTRODUCTION:

This Project Report card management System includes the facilities of registration, modification, search, display, Deletion etc. of students information about marks, their names and roll number.

This program searches for a record information on the basis of the roll number of that student.

This software is very useful for small schools for maintaining their report card records and that is in cost effective way.

MODULES:-

- main:- To control all other functions flow
- entryMenu:- To display the the starting navigation menu
- acceptdata:- To write information of student in file
- view_specific:- To display the data of specific student
- deleterecord:- To Delete the any students full information
- modify:- To modify record of specific student
- result:- To display the result Menu

PROJECT OBJECTIVES

- The objective of developing such a computerization system is to reduce the paperwork and time in Report Card management thereby increasing the efficiency and decreasing the workload.
- To use fundamental Data Structures, Classes and Objects to implement member functions that register, search, display, modify, delete student information about the marks and their names and roll number.

SOFTWARE AND HARDWARE REQUIREMENTS :

- Printer : To print project report
- Compiler : Dev C++, Code Blocks
- Operating System: Windows xp, 7,8,10.
- RAM : 512 MB or more than 512 MB
- Processor : Dual core
- Hard Disk Usage : 1 to 5 MB

THEORY OF CONCEPTS USED

Input /Output with files

C++ provides the following classes to perform output and input of characters to/from files:

S.No	Data Type & Description
1	**Ofstream** *This data type shows the output file stream and used to create files and to write information to files.*
2	**ifstream** *This data type shows the input file stream and is used to read information from files.*
3	**fstream** *This data type represents the file stream generally, and has the capabilities of both ofstream and ifstream which means it can create files, write information to files, and read information from files.*

All C++ compilers come with classes for streaming input from the console and output to the console. These classes are defined by putting the directive #include <iostream> at the top of the code. The istream class has methods for detecting input errors and the end of input data. The ostream class has methods for formatting output, i.e. specifying scientific notation, fixed decimal notation, or a combination thereof, and for specifying the number of decimal digits displayed. Using some of the features of these classes, we add the capability of reading and writing our own custom types. Finally, the ifstream and ofstream classes let us read from and write to named files.

Opening a File

A file must be opened before you can read from it or write to it. Either **ofstream** or **fstream** object may be used to open a file for writing. And ifstream object is used to open a file for reading purpose only.

Following is the standard syntax for open() function, which is a member of fstream, ifstream, and ofstream objects.

```
void open(const char *filename, ios::opening mode);
```

Here, the first argument specifies the name and location of the file to be opened and the second argument of the **open()** member function defines the mode in which the file should be opened.

Sr.No	Mode Flag & Description
1	**ios::app** *Append mode. All output to that file to be appended to the end.*
2	**ios::binary** *Open a file for storing in such a way that it is not in readable format by human*
3	**ios::in** *Open a file for reading.*
4	**ios::out** *Open a file for writing.*
5	**ios::trunc** *If the file already exists, its contents will be truncated before opening the file.*

You can combine two or more of these values by **OR**ing them together. For example if you want to open a file in write mode and want to truncate it in case that already exists, **following will be the syntax –**

```
ofstream outfile;
outfile.open("file.dat", ios::app || ios::binary );
```

Similar way, you can open a file for reading and writing purpose as follows −

```
fstream  afile;
afile.open("file.dat", ios::out || ios::in );
```

Closing a File

When a C++ program terminates it automatically flushes all the streams, release all the allocated memory and close all the opened files. But it is always a good practice that a programmer should close all the opened files before program termination.

Following is the syntax for close() function

```
void close();
```

Writing to a File

While doing C++ programming, you write information to a file from your program using the stream insertion operator (<<) just as you use that operator to output information to the screen. The only difference is that you use an **ofstream** or **fstream** object instead of the **cout** object.

Reading from a File

You read information from a file into your program using the stream extraction operator (>>) just as you use that operator to input information from the keyboard. The only difference is that you use an **ifstream** or **fstream** object instead of the **cin** object.

ostream::seekp(pos)

The **seekp(pos)** method of **ostream in C++** is used to set the position of the pointer in the output sequence with the specified position. This method takes the new position to be set and returns this ostream instance with the position set to the specified new position.

Syntax:

```
ostream& seekp(streampos pos);
```

Parameter: This method takes the **new position** to be set as the parameter.

Return Value: This method returns **this ostream instance** with the position set to the specified new position.

Ignore()

Essentially, for `std::cin` statements you use ignore before you do a `getline` call, because when a user inputs something with `std::cin`, they hit enter and a `'\n'` char gets into the `cin` buffer. Then if you use `getline`, it gets the newline char instead of the string you want.

So you do a `std::cin.ignore(1000,'\n')` and that should clear the buffer up to the string that you want.

(The 1000 is put there to skip over a specific amount of chars before the specified break point, in this case, the \n newline character.)

Defining color of output

For background and foreground, type in a number from 0 - 9 or a letter from A - F.

```
system("color A1");
std::cout<<"hi"<<std::endl;
```

That would display the letters "hi" with a green background and blue text.

Sleep()

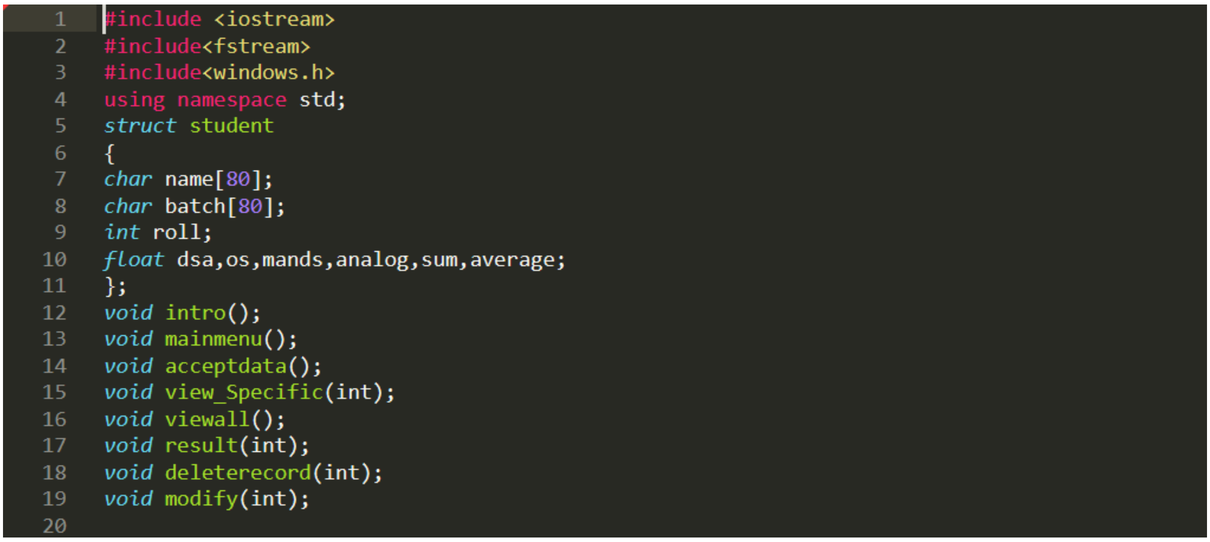
This function stops the program running for the time given in bracket.

```
Sleep(x);
```

This stops the program for x milliseconds.

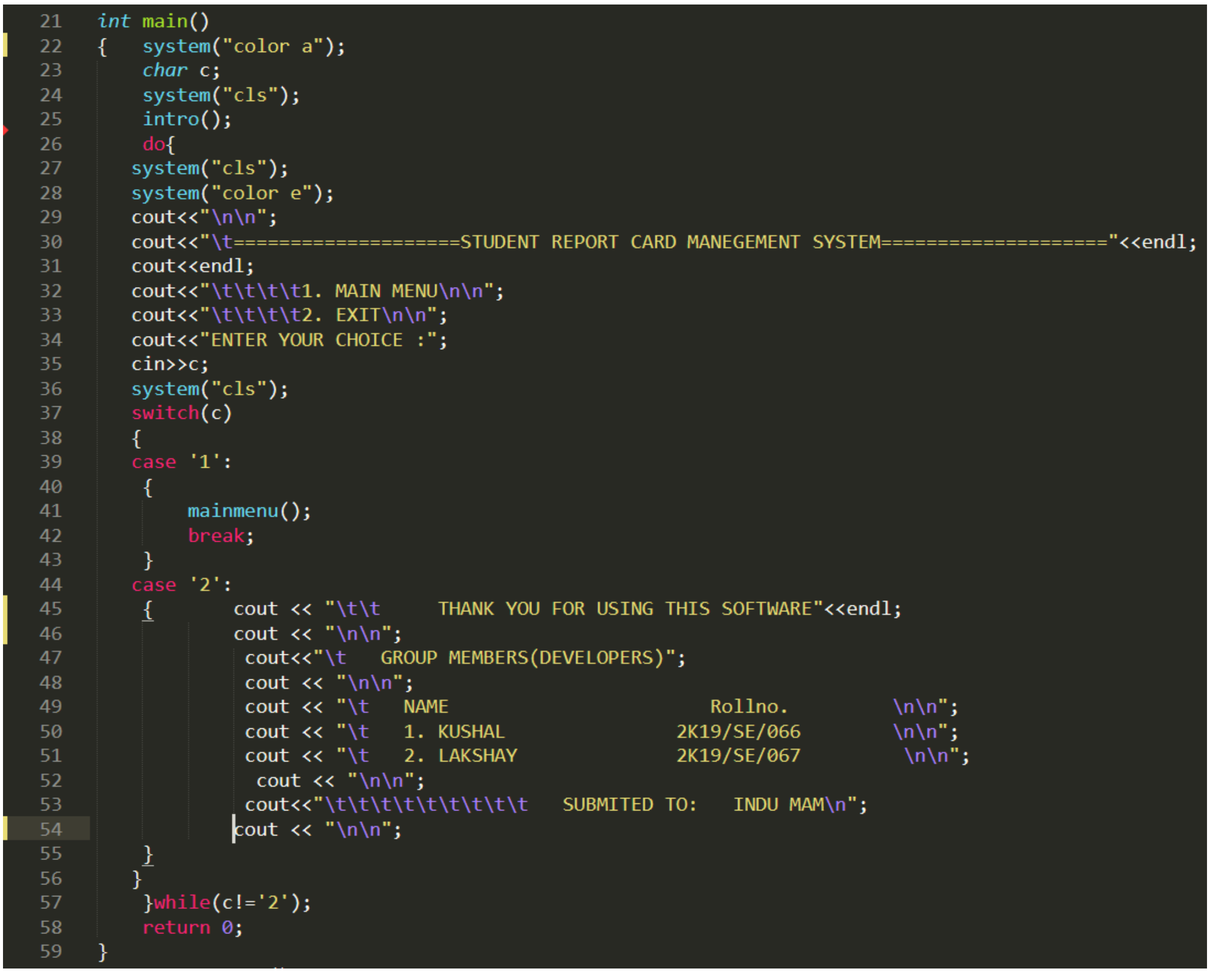
SOURCE CODE

Header files and variables



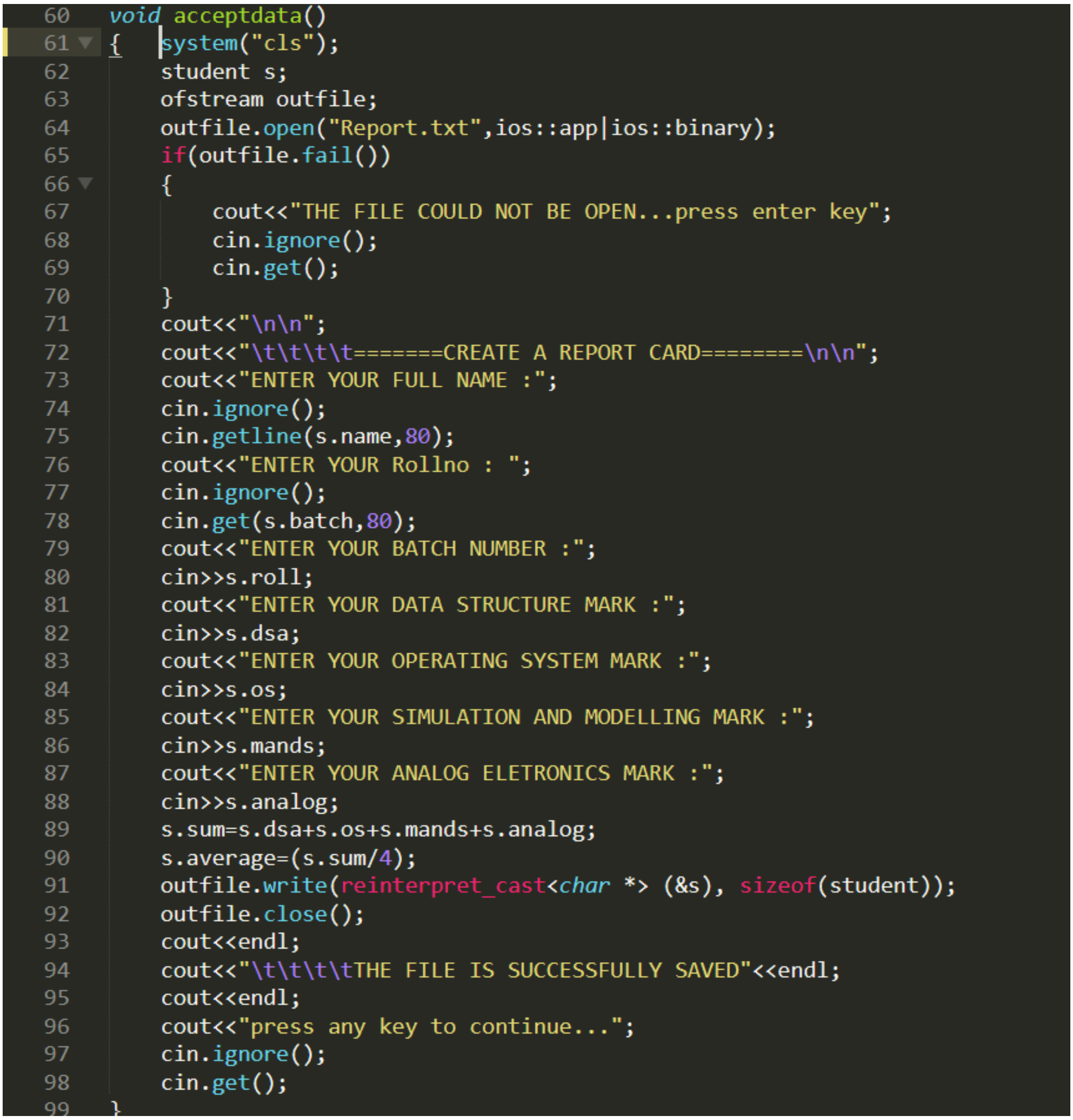
```
#include <iostream>
#include<fstream>
#include<windows.h>
using namespace std;
struct student
{
char name[80];
char batch[80];
int roll;
float dsa,os,mands,analog,sum,average;
};
void intro();
void mainmenu();
void acceptdata();
void view_Specific(int);
void viewall();
void result(int);
void deleterecord(int);
void modify(int);
```

THE MAIN FUNCTION



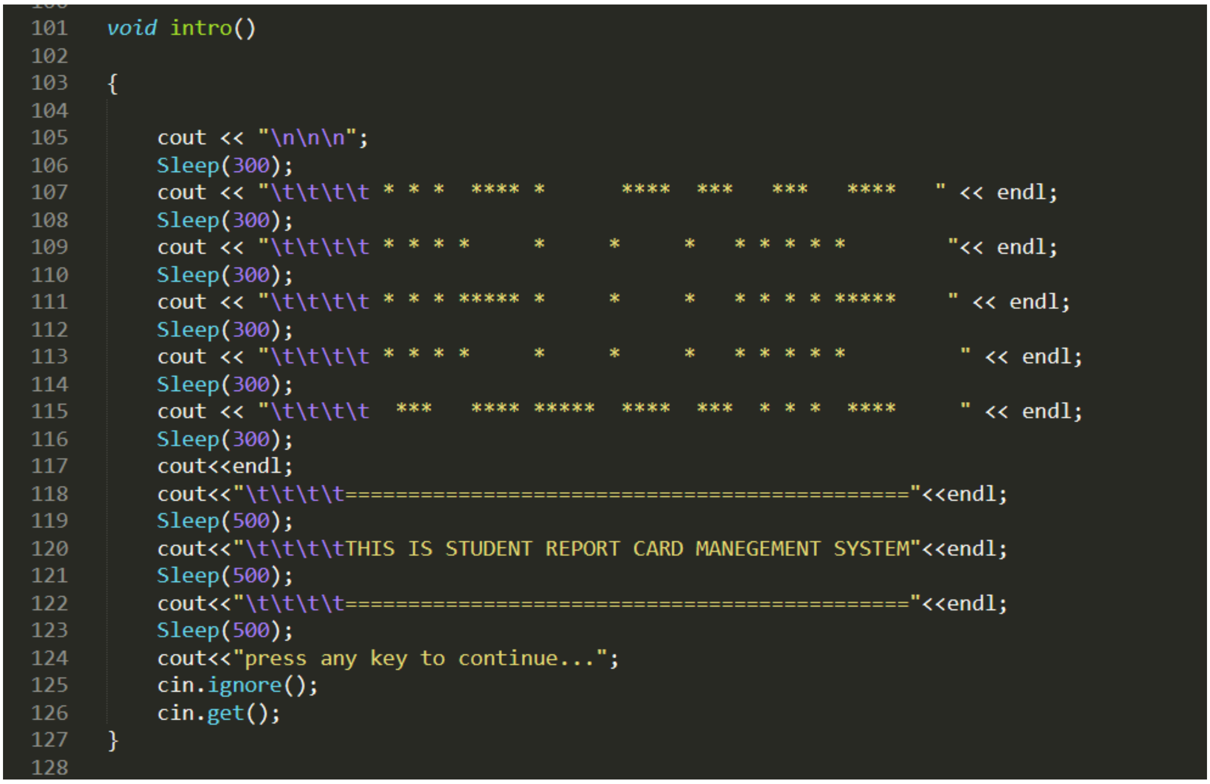
```
int main()
{   system("color a");
    char c;
    system("cls");
    intro();
    do{
   system("cls");
   system("color e");
   cout<<"\n\n";
   cout<<"\t====================STUDENT REPORT CARD MANEGEMENT SYSTEM===================="<<endl;
   cout<<endl;
   cout<<"\t\t\t\t1. MAIN MENU\n\n";
   cout<<"\t\t\t\t2. EXIT\n\n";
   cout<<"ENTER YOUR CHOICE :";
   cin>>c;
   system("cls");
   switch(c)
   {
   case '1':
    {
        mainmenu();
        break;
    }
   case '2':
    {       cout << "\t\t     THANK YOU FOR USING THIS SOFTWARE"<<endl;
            cout << "\n\n";
             cout<<"\t   GROUP MEMBERS(DEVELOPERS)";
             cout << "\n\n";
             cout << "\t   NAME                       Rollno.         \n\n";
             cout << "\t   1. KUSHAL               2K19/SE/066        \n\n";
             cout << "\t   2. LAKSHAY              2K19/SE/067         \n\n";
              cout << "\n\n";
             cout<<"\t\t\t\t\t\t\t\t\t   SUBMITED TO:   INDU MAM\n";
            cout << "\n\n";
    }
   }
    }while(c!='2');
    return 0;
}
```

FUNCTION FOR ACCEPTING DATA



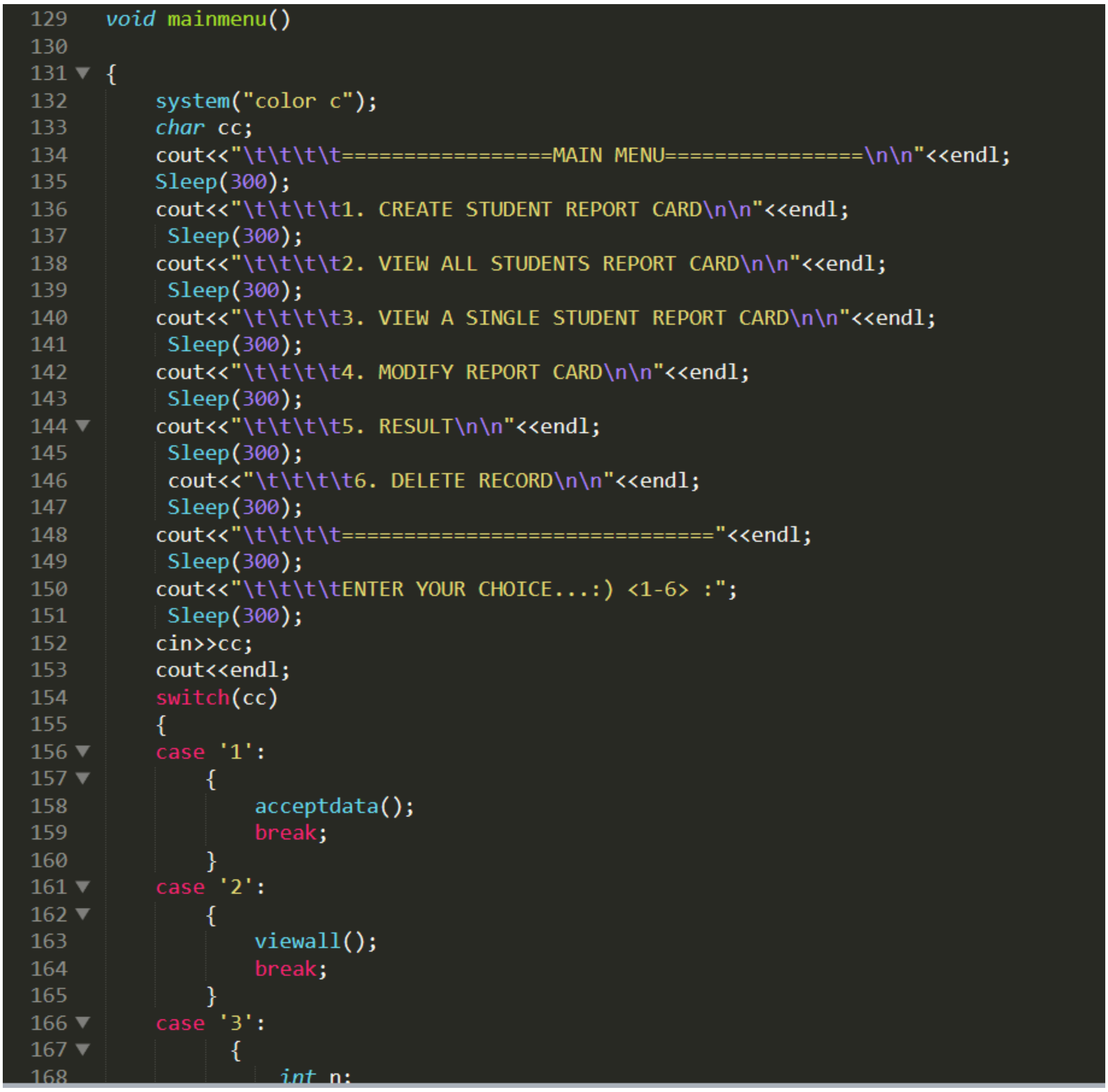
```
void acceptdata()
{   system("cls");
    student s;
    ofstream outfile;
    outfile.open("Report.txt",ios::app|ios::binary);
    if(outfile.fail())
    {
        cout<<"THE FILE COULD NOT BE OPEN...press enter key";
        cin.ignore();
        cin.get();
    }
    cout<<"\n\n";
    cout<<"\t\t\t\t=======CREATE A REPORT CARD========\n\n";
    cout<<"ENTER YOUR FULL NAME :";
    cin.ignore();
    cin.getline(s.name,80);
    cout<<"ENTER YOUR Rollno : ";
    cin.ignore();
    cin.get(s.batch,80);
    cout<<"ENTER YOUR BATCH NUMBER :";
    cin>>s.roll;
    cout<<"ENTER YOUR DATA STRUCTURE MARK :";
    cin>>s.dsa;
    cout<<"ENTER YOUR OPERATING SYSTEM MARK :";
    cin>>s.os;
    cout<<"ENTER YOUR SIMULATION AND MODELLING MARK :";
    cin>>s.mands;
    cout<<"ENTER YOUR ANALOG ELETRONICS MARK :";
    cin>>s.analog;
    s.sum=s.dsa+s.os+s.mands+s.analog;
    s.average=(s.sum/4);
    outfile.write(reinterpret_cast<char *> (&s), sizeof(student));
    outfile.close();
    cout<<endl;
    cout<<"\t\t\t\tTHE FILE IS SUCCESSFULLY SAVED"<<endl;
    cout<<endl;
    cout<<"press any key to continue...";
    cin.ignore();
    cin.get();
}
```

FUNCTION FOR THE INTRODUCTION

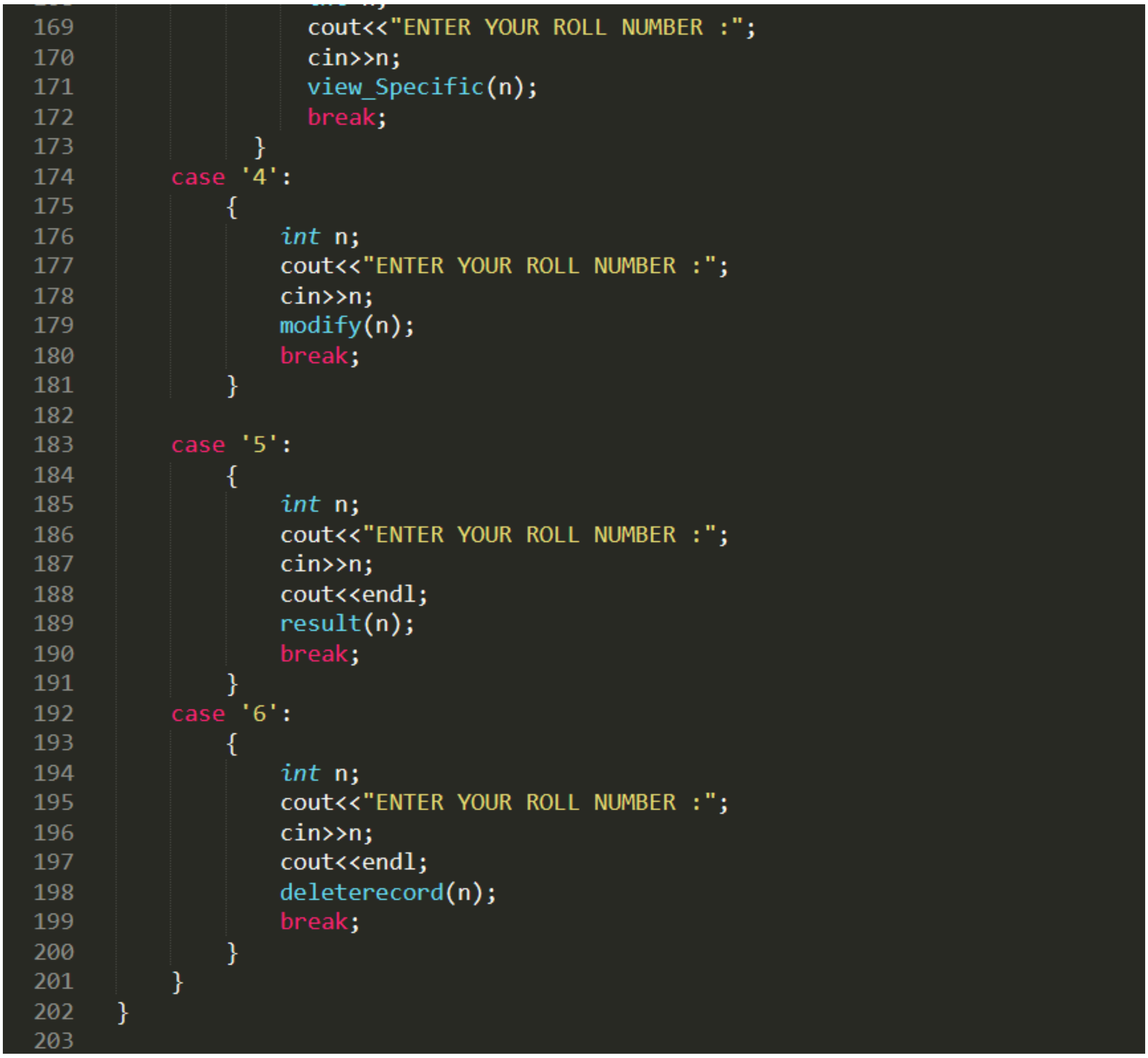


```cpp
void intro()

{

    cout << "\n\n\n";
    Sleep(300);
    cout << "\t\t\t\t * * *  **** *      ****  ***   ***   ****   " << endl;
    Sleep(300);
    cout << "\t\t\t\t * * * *     *     *     *   * * * * *        "<< endl;
    Sleep(300);
    cout << "\t\t\t\t * * * ***** *     *     *   * * * * *****    " << endl;
    Sleep(300);
    cout << "\t\t\t\t * * * *     *     *     *   * * * * *         " << endl;
    Sleep(300);
    cout << "\t\t\t\t  ***   **** *****  ****  ***  * * *  ****     " << endl;
    Sleep(300);
    cout<<endl;
    cout<<"\t\t\t\t============================================="<<endl;
    Sleep(500);
    cout<<"\t\t\t\tTHIS IS STUDENT REPORT CARD MANEGEMENT SYSTEM"<<endl;
    Sleep(500);
    cout<<"\t\t\t\t============================================="<<endl;
    Sleep(500);
    cout<<"press any key to continue...";
    cin.ignore();
    cin.get();
}
```

FUNCTION FOR MAIN MENU

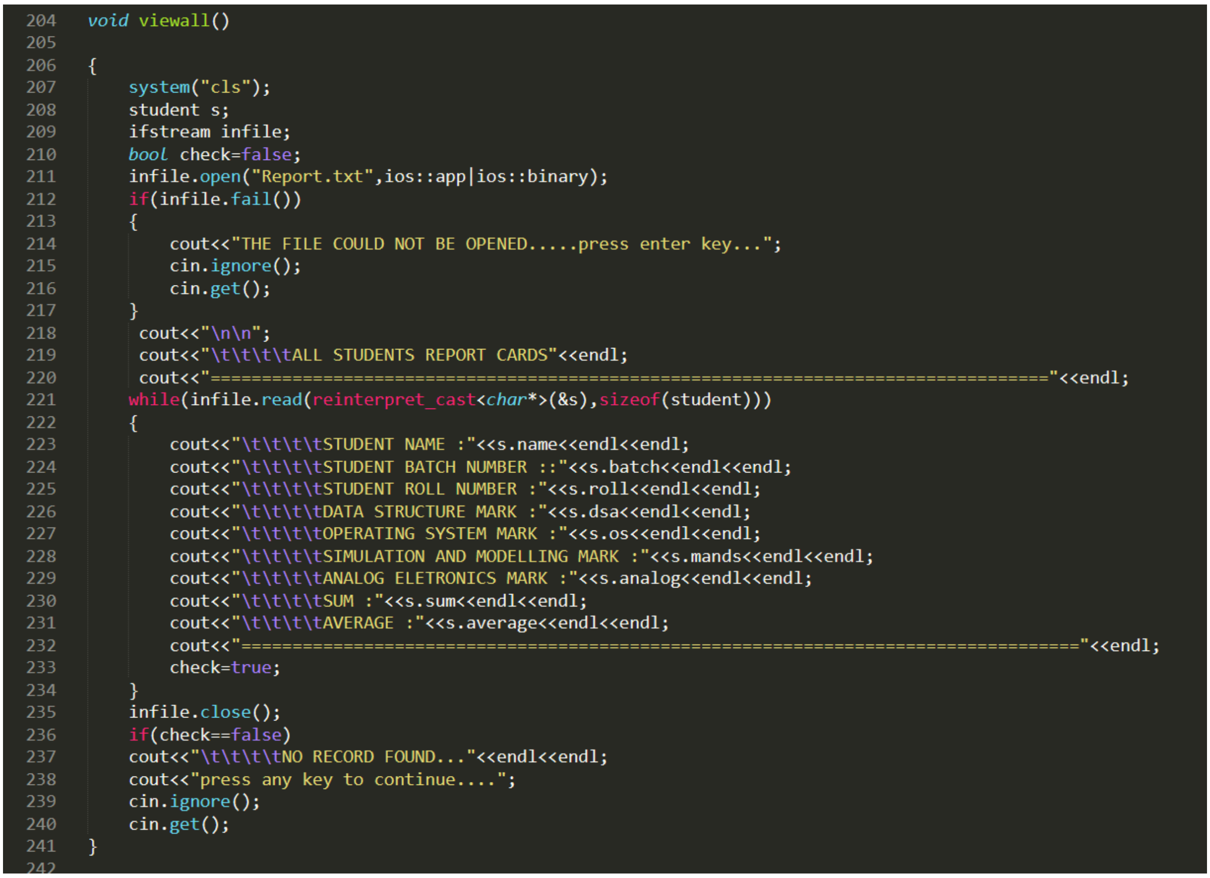


```
void mainmenu()

{
    system("color c");
    char cc;
    cout<<"\t\t\t\t=================MAIN MENU================\n\n"<<endl;
    Sleep(300);
    cout<<"\t\t\t\t1. CREATE STUDENT REPORT CARD\n\n"<<endl;
     Sleep(300);
    cout<<"\t\t\t\t2. VIEW ALL STUDENTS REPORT CARD\n\n"<<endl;
     Sleep(300);
    cout<<"\t\t\t\t3. VIEW A SINGLE STUDENT REPORT CARD\n\n"<<endl;
     Sleep(300);
    cout<<"\t\t\t\t4. MODIFY REPORT CARD\n\n"<<endl;
     Sleep(300);
    cout<<"\t\t\t\t5. RESULT\n\n"<<endl;
     Sleep(300);
     cout<<"\t\t\t\t6. DELETE RECORD\n\n"<<endl;
     Sleep(300);
    cout<<"\t\t\t\t==============================="<<endl;
     Sleep(300);
    cout<<"\t\t\t\tENTER YOUR CHOICE...:) <1-6> :";
     Sleep(300);
    cin>>cc;
    cout<<endl;
    switch(cc)
    {
    case '1':
        {
            acceptdata();
            break;
        }
    case '2':
        {
            viewall();
            break;
        }
    case '3':
          {
              int n;
```



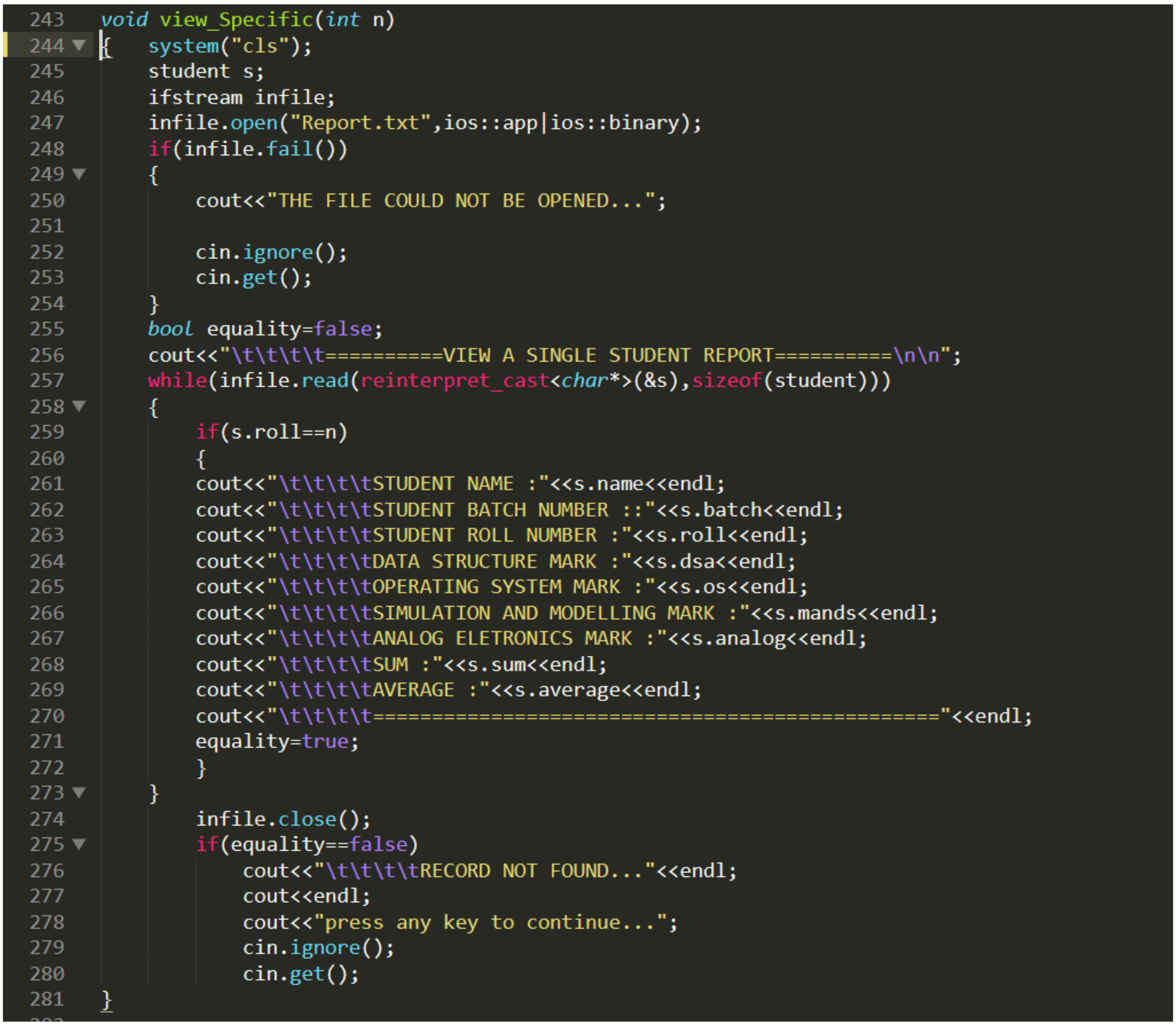
```
              cout<<"ENTER YOUR ROLL NUMBER :";
              cin>>n;
              view_Specific(n);
              break;
          }
    case '4':
        {
            int n;
            cout<<"ENTER YOUR ROLL NUMBER :";
            cin>>n;
            modify(n);
            break;
        }

    case '5':
        {
            int n;
            cout<<"ENTER YOUR ROLL NUMBER :";
            cin>>n;
            cout<<endl;
            result(n);
            break;
        }
    case '6':
        {
            int n;
            cout<<"ENTER YOUR ROLL NUMBER :";
            cin>>n;
            cout<<endl;
            deleterecord(n);
            break;
        }
    }
}
```

FUNCTION FOR DISPLAYING ALL RECORDS



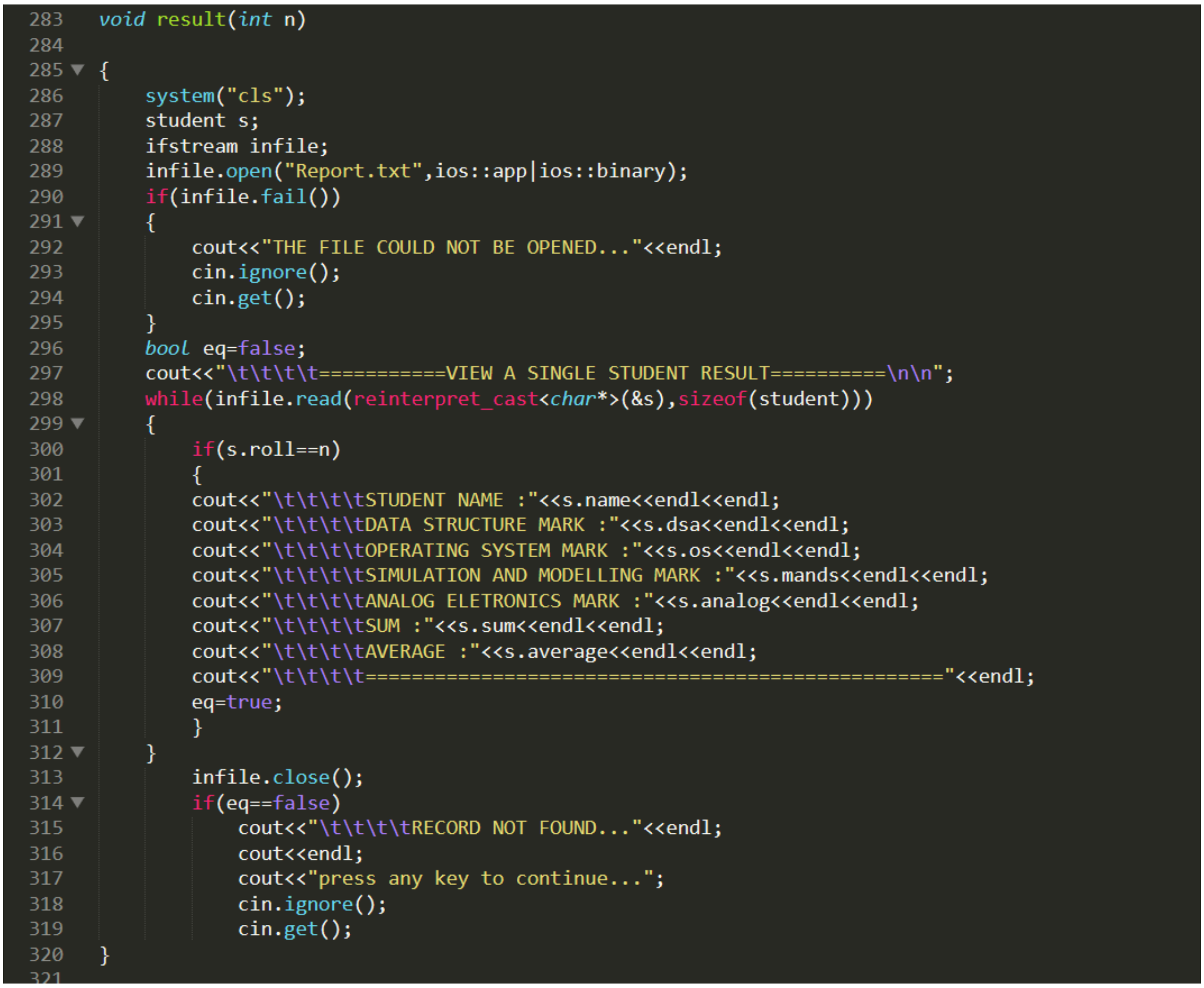
```
void viewall()

{
    system("cls");
    student s;
    ifstream infile;
    bool check=false;
    infile.open("Report.txt",ios::app|ios::binary);
    if(infile.fail())
    {
        cout<<"THE FILE COULD NOT BE OPENED.....press enter key...";
        cin.ignore();
        cin.get();
    }
     cout<<"\n\n";
     cout<<"\t\t\t\tALL STUDENTS REPORT CARDS"<<endl;
     cout<<"=================================================================================="<<endl;
    while(infile.read(reinterpret_cast<char*>(&s),sizeof(student)))
    {
        cout<<"\t\t\t\tSTUDENT NAME :"<<s.name<<endl<<endl;
        cout<<"\t\t\t\tSTUDENT BATCH NUMBER ::"<<s.batch<<endl<<endl;
        cout<<"\t\t\t\tSTUDENT ROLL NUMBER :"<<s.roll<<endl<<endl;
        cout<<"\t\t\t\tDATA STRUCTURE MARK :"<<s.dsa<<endl<<endl;
        cout<<"\t\t\t\tOPERATING SYSTEM MARK :"<<s.os<<endl<<endl;
        cout<<"\t\t\t\tSIMULATION AND MODELLING MARK :"<<s.mands<<endl<<endl;
        cout<<"\t\t\t\tANALOG ELETRONICS MARK :"<<s.analog<<endl<<endl;
        cout<<"\t\t\t\tSUM :"<<s.sum<<endl<<endl;
        cout<<"\t\t\t\tAVERAGE :"<<s.average<<endl<<endl;
        cout<<"=================================================================================="<<endl;
        check=true;
    }
    infile.close();
    if(check==false)
    cout<<"\t\t\t\tNO RECORD FOUND..."<<endl<<endl;
    cout<<"press any key to continue....";
    cin.ignore();
    cin.get();
}
```

FUNCTION FOR DISPLAYING SPECIFIC RECORD



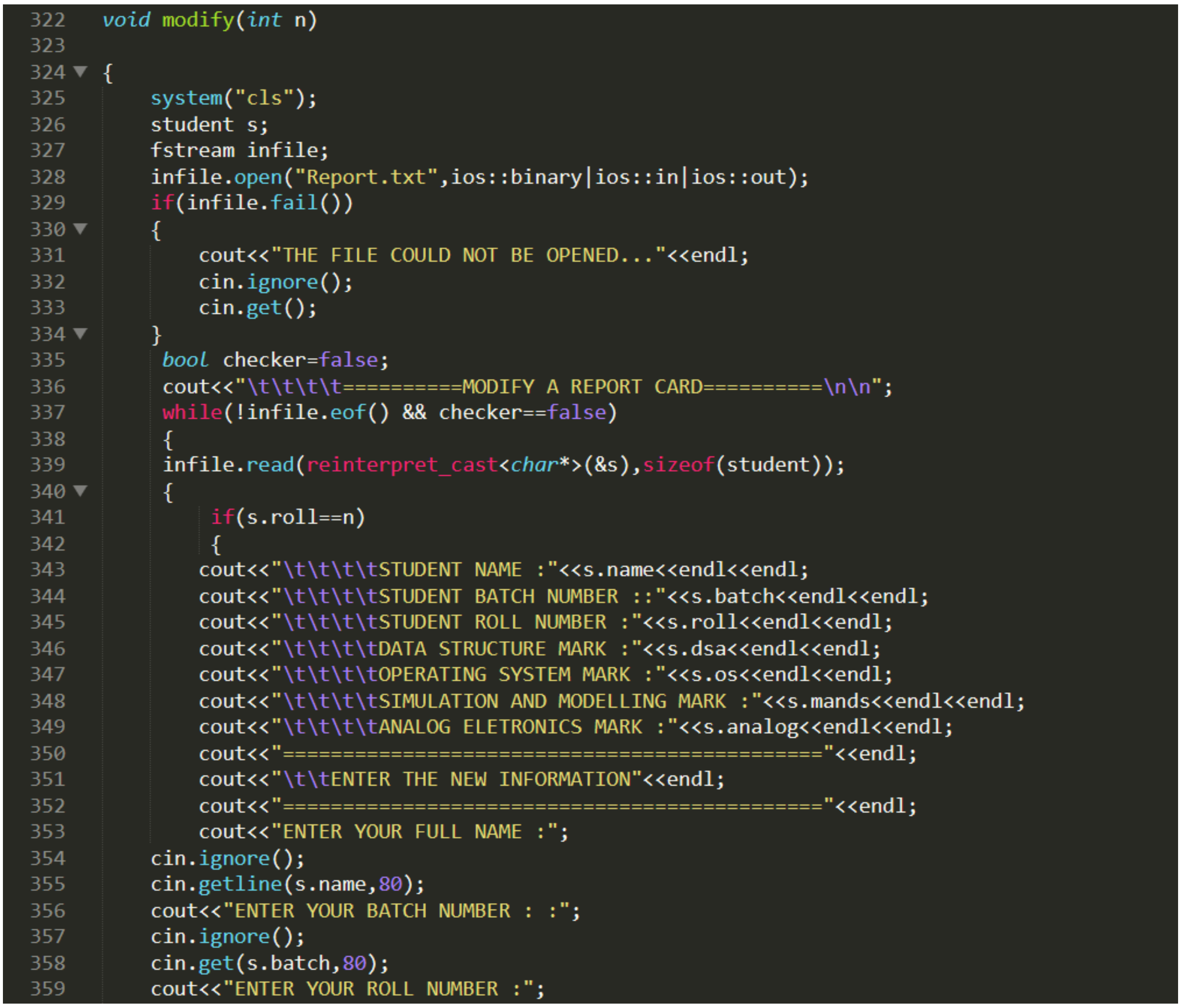
```cpp
void view_Specific(int n)
{   system("cls");
    student s;
    ifstream infile;
    infile.open("Report.txt",ios::app|ios::binary);
    if(infile.fail())
    {
        cout<<"THE FILE COULD NOT BE OPENED...";

        cin.ignore();
        cin.get();
    }
    bool equality=false;
    cout<<"\t\t\t\t==========VIEW A SINGLE STUDENT REPORT==========\n\n";
    while(infile.read(reinterpret_cast<char*>(&s),sizeof(student)))
    {
        if(s.roll==n)
        {
        cout<<"\t\t\t\tSTUDENT NAME :"<<s.name<<endl;
        cout<<"\t\t\t\tSTUDENT BATCH NUMBER ::"<<s.batch<<endl;
        cout<<"\t\t\t\tSTUDENT ROLL NUMBER :"<<s.roll<<endl;
        cout<<"\t\t\t\tDATA STRUCTURE MARK :"<<s.dsa<<endl;
        cout<<"\t\t\t\tOPERATING SYSTEM MARK :"<<s.os<<endl;
        cout<<"\t\t\t\tSIMULATION AND MODELLING MARK :"<<s.mands<<endl;
        cout<<"\t\t\t\tANALOG ELETRONICS MARK :"<<s.analog<<endl;
        cout<<"\t\t\t\tSUM :"<<s.sum<<endl;
        cout<<"\t\t\t\tAVERAGE :"<<s.average<<endl;
        cout<<"\t\t\t\t================================================"<<endl;
        equality=true;
        }
    }
        infile.close();
        if(equality==false)
            cout<<"\t\t\t\tRECORD NOT FOUND..."<<endl;
            cout<<endl;
            cout<<"press any key to continue...";
            cin.ignore();
            cin.get();
}
```

FUNCTION FOR RESULT OF THE STUDENT

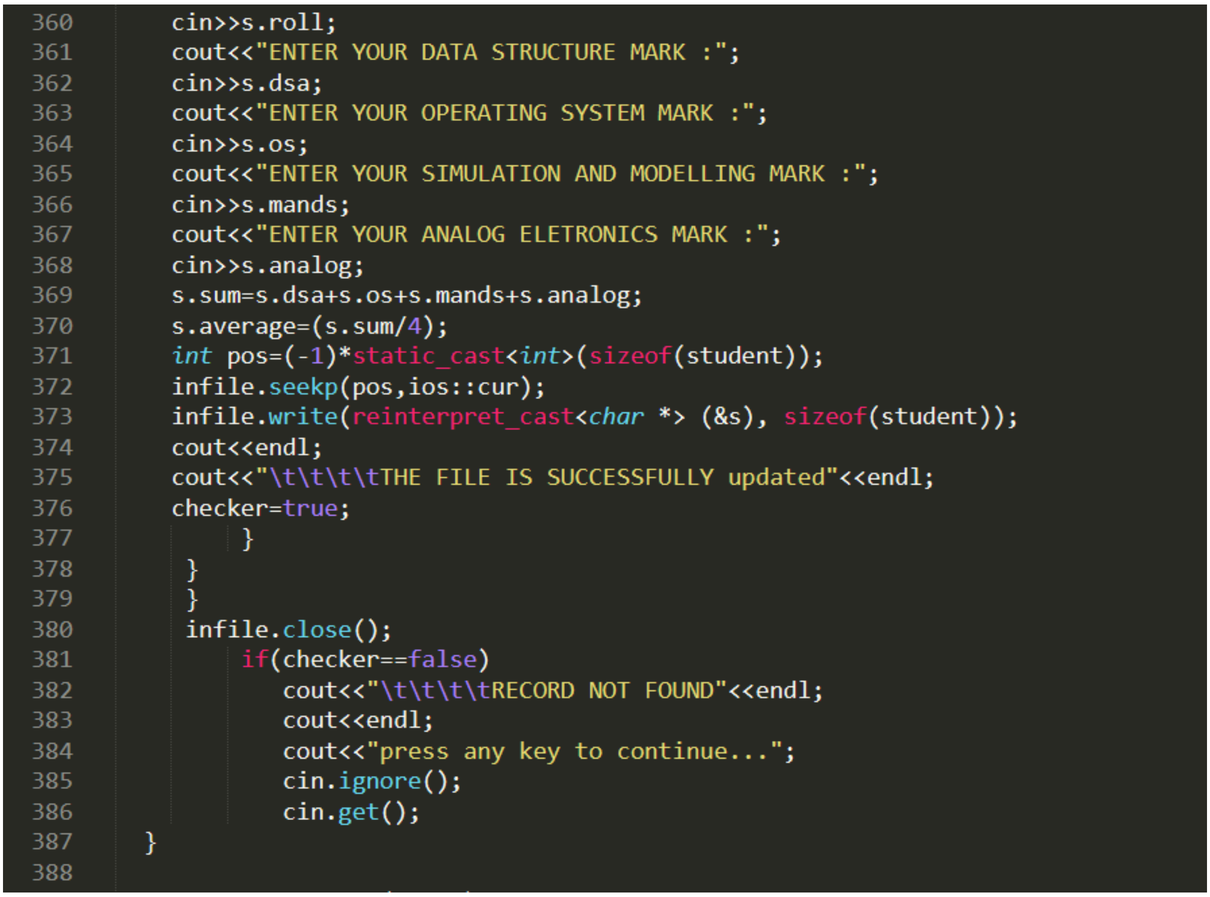


```
void result(int n)

{
    system("cls");
    student s;
    ifstream infile;
    infile.open("Report.txt",ios::app|ios::binary);
    if(infile.fail())
    {
        cout<<"THE FILE COULD NOT BE OPENED..."<<endl;
        cin.ignore();
        cin.get();
    }
    bool eq=false;
    cout<<"\t\t\t\t===========VIEW A SINGLE STUDENT RESULT==========\n\n";
    while(infile.read(reinterpret_cast<char*>(&s),sizeof(student)))
    {
        if(s.roll==n)
        {
        cout<<"\t\t\t\tSTUDENT NAME :"<<s.name<<endl<<endl;
        cout<<"\t\t\t\tDATA STRUCTURE MARK :"<<s.dsa<<endl<<endl;
        cout<<"\t\t\t\tOPERATING SYSTEM MARK :"<<s.os<<endl<<endl;
        cout<<"\t\t\t\tSIMULATION AND MODELLING MARK :"<<s.mands<<endl<<endl;
        cout<<"\t\t\t\tANALOG ELETRONICS MARK :"<<s.analog<<endl<<endl;
        cout<<"\t\t\t\tSUM :"<<s.sum<<endl<<endl;
        cout<<"\t\t\t\tAVERAGE :"<<s.average<<endl<<endl;
        cout<<"\t\t\t\t=================================================="<<endl;
        eq=true;
        }
    }
        infile.close();
        if(eq==false)
            cout<<"\t\t\t\tRECORD NOT FOUND..."<<endl;
            cout<<endl;
            cout<<"press any key to continue...";
            cin.ignore();
            cin.get();
}
```

FUNCTION FOR MODIFYING STUDENT RECORD

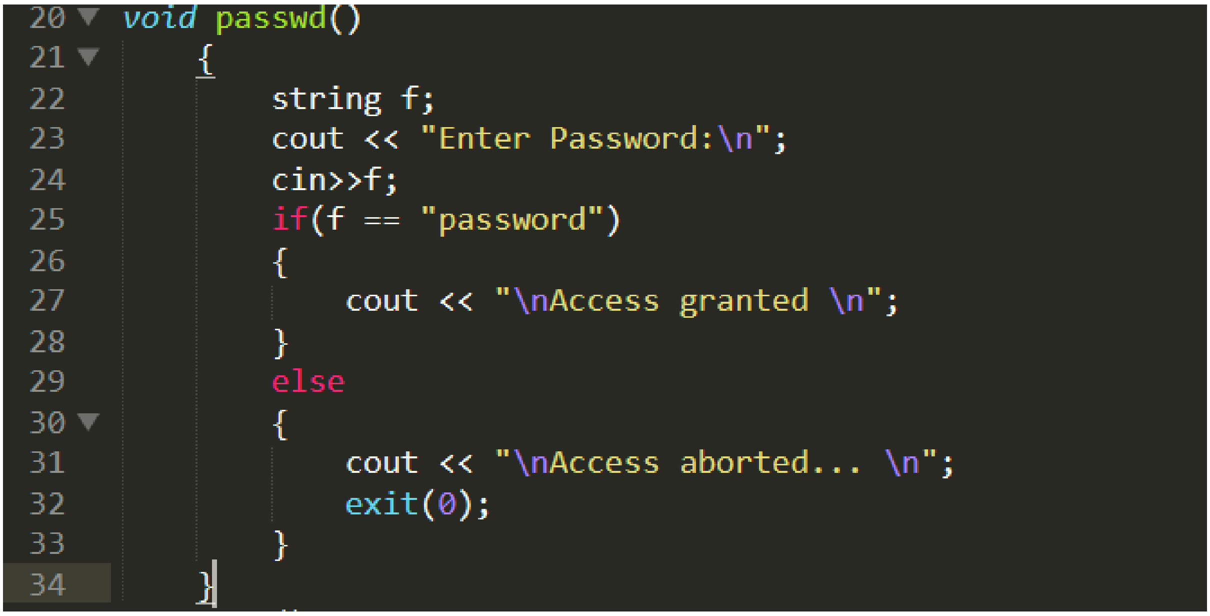


```cpp
void modify(int n)

{
    system("cls");
    student s;
    fstream infile;
    infile.open("Report.txt",ios::binary|ios::in|ios::out);
    if(infile.fail())
    {
        cout<<"THE FILE COULD NOT BE OPENED..."<<endl;
        cin.ignore();
        cin.get();
    }
     bool checker=false;
     cout<<"\t\t\t\t==========MODIFY A REPORT CARD==========\n\n";
     while(!infile.eof() && checker==false)
     {
     infile.read(reinterpret_cast<char*>(&s),sizeof(student));
     {
         if(s.roll==n)
         {
        cout<<"\t\t\t\tSTUDENT NAME :"<<s.name<<endl<<endl;
        cout<<"\t\t\t\tSTUDENT BATCH NUMBER ::"<<s.batch<<endl<<endl;
        cout<<"\t\t\t\tSTUDENT ROLL NUMBER :"<<s.roll<<endl<<endl;
        cout<<"\t\t\t\tDATA STRUCTURE MARK :"<<s.dsa<<endl<<endl;
        cout<<"\t\t\t\tOPERATING SYSTEM MARK :"<<s.os<<endl<<endl;
        cout<<"\t\t\t\tSIMULATION AND MODELLING MARK :"<<s.mands<<endl<<endl;
        cout<<"\t\t\t\tANALOG ELETRONICS MARK :"<<s.analog<<endl<<endl;
        cout<<"============================================="<<endl;
        cout<<"\t\tENTER THE NEW INFORMATION"<<endl;
        cout<<"============================================="<<endl;
        cout<<"ENTER YOUR FULL NAME :";
    cin.ignore();
    cin.getline(s.name,80);
    cout<<"ENTER YOUR BATCH NUMBER : :";
    cin.ignore();
    cin.get(s.batch,80);
    cout<<"ENTER YOUR ROLL NUMBER :";
```



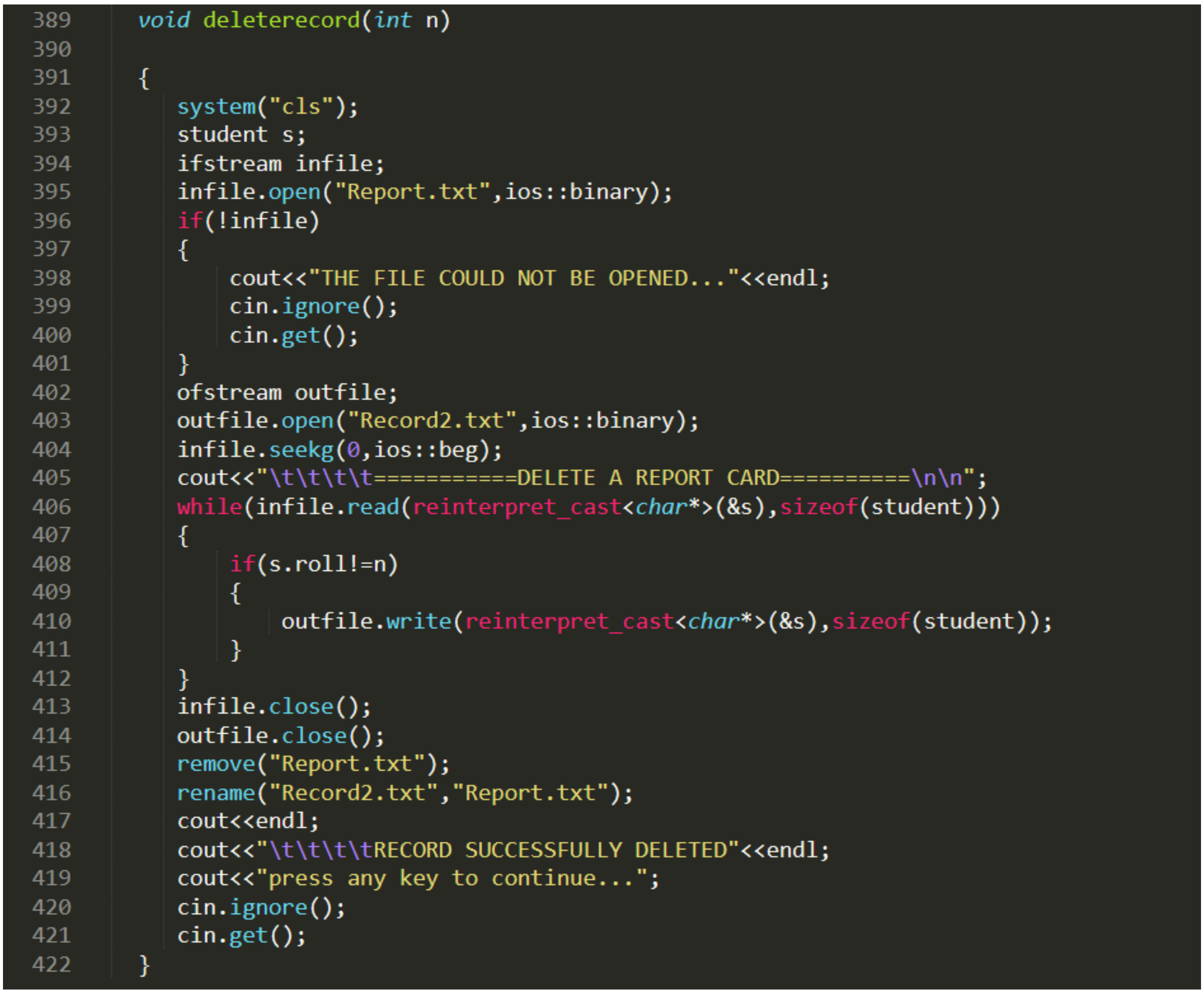
```
  cin>>s.roll;
  cout<<"ENTER YOUR DATA STRUCTURE MARK :";
  cin>>s.dsa;
  cout<<"ENTER YOUR OPERATING SYSTEM MARK :";
  cin>>s.os;
  cout<<"ENTER YOUR SIMULATION AND MODELLING MARK :";
  cin>>s.mands;
  cout<<"ENTER YOUR ANALOG ELETRONICS MARK :";
  cin>>s.analog;
  s.sum=s.dsa+s.os+s.mands+s.analog;
  s.average=(s.sum/4);
  int pos=(-1)*static_cast<int>(sizeof(student));
  infile.seekp(pos,ios::cur);
  infile.write(reinterpret_cast<char *> (&s), sizeof(student));
  cout<<endl;
  cout<<"\t\t\t\tTHE FILE IS SUCCESSFULLY updated"<<endl;
  checker=true;
       }
   }
   }
   infile.close();
       if(checker==false)
          cout<<"\t\t\t\tRECORD NOT FOUND"<<endl;
          cout<<endl;
          cout<<"press any key to continue...";
          cin.ignore();
          cin.get();
}

```

PASSWORD FOR MODIFYING STUDENT RECORD



```
void passwd()
    {
        string f;
        cout << "Enter Password:\n";
        cin>>f;
        if(f == "password")
        {
            cout << "\nAccess granted \n";
        }
        else
        {
            cout << "\nAccess aborted... \n";
            exit(0);
        }
    }
```

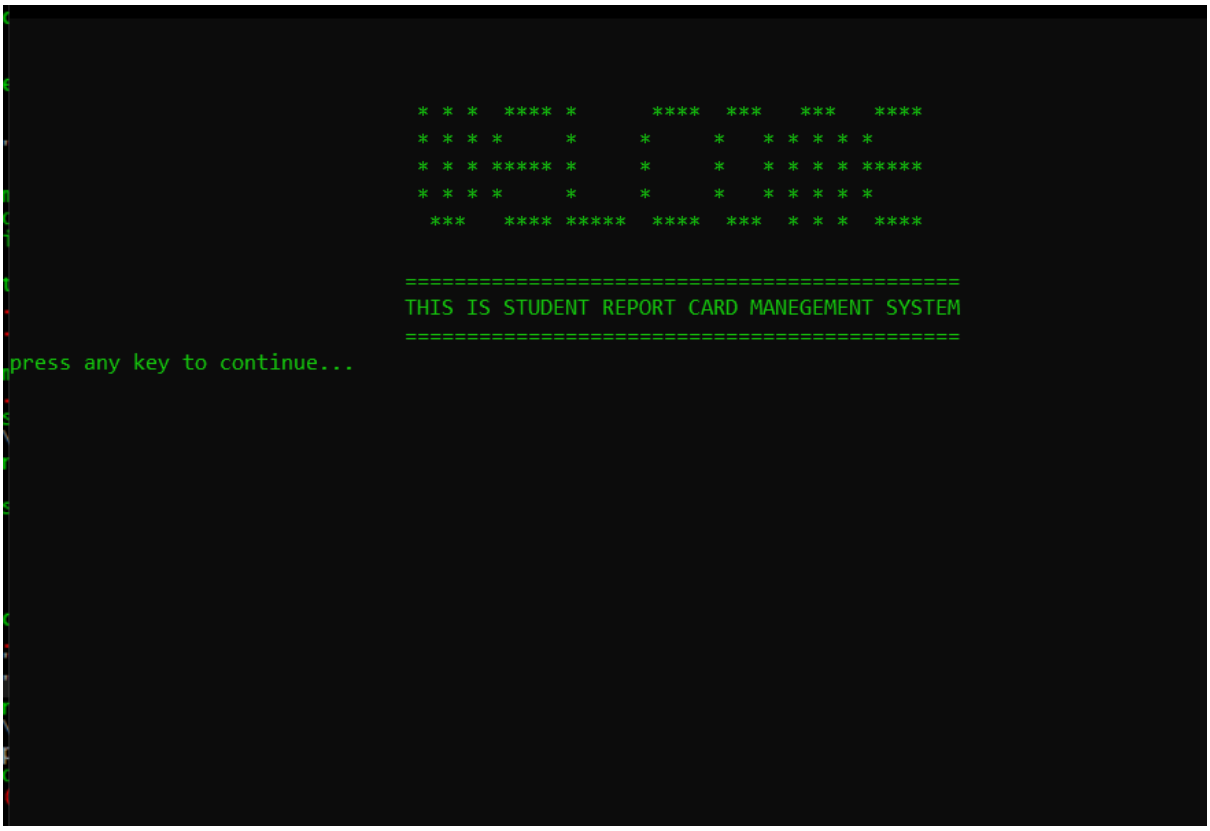
FUNCTION FOR DELETING RECORD



```cpp
void deleterecord(int n)

{
    system("cls");
    student s;
    ifstream infile;
    infile.open("Report.txt",ios::binary);
    if(!infile)
    {
        cout<<"THE FILE COULD NOT BE OPENED..."<<endl;
        cin.ignore();
        cin.get();
    }
    ofstream outfile;
    outfile.open("Record2.txt",ios::binary);
    infile.seekg(0,ios::beg);
    cout<<"\t\t\t\t===========DELETE A REPORT CARD==========\n\n";
    while(infile.read(reinterpret_cast<char*>(&s),sizeof(student)))
    {
        if(s.roll!=n)
        {
            outfile.write(reinterpret_cast<char*>(&s),sizeof(student));
        }
    }
    infile.close();
    outfile.close();
    remove("Report.txt");
    rename("Record2.txt","Report.txt");
    cout<<endl;
    cout<<"\t\t\t\tRECORD SUCCESSFULLY DELETED"<<endl;
    cout<<"press any key to continue...";
    cin.ignore();
    cin.get();
}
```

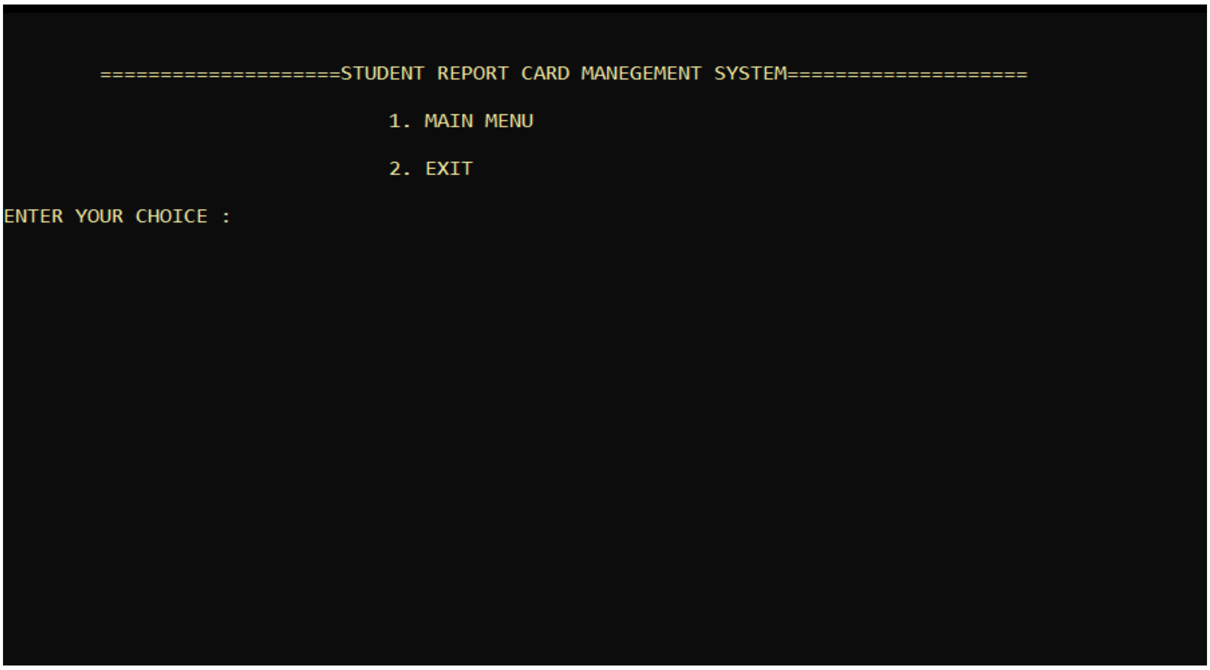
OUTPUT SCREENS

OUTPUT FOR THE INTRODUCTION



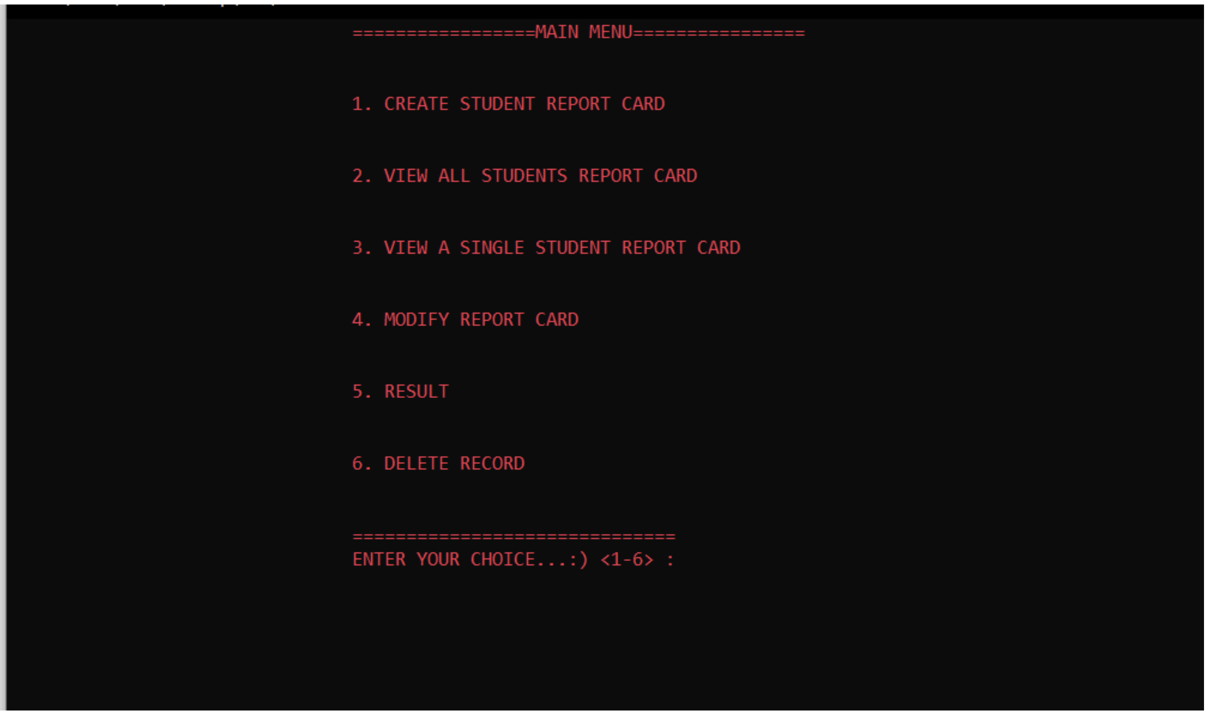
```
                                 * * *  **** *      ****  ***   ***   ****
                                 * * * *     *     *     *   * * * * *
                                 * * * ***** *     *     *   * * * * *****
                                 * * * *     *     *     *   * * * * *
                                  ***   **** *****  ****  ***  * * *  ****

                                =============================================
                                THIS IS STUDENT REPORT CARD MANEGEMENT SYSTEM
                                =============================================
press any key to continue...
```

OUTPUT FOR CHOOSING OPTION



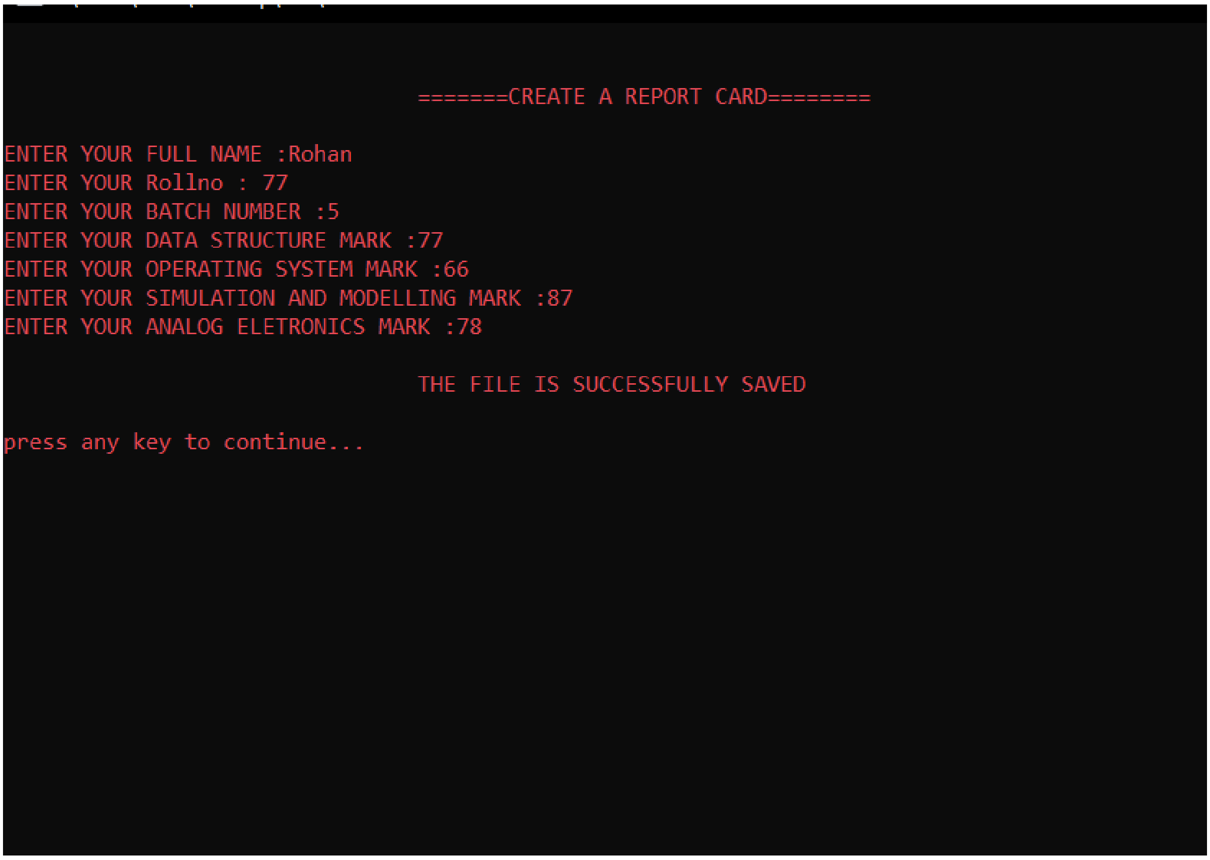
```
        ====================STUDENT REPORT CARD MANEGEMENT SYSTEM====================

                                1. MAIN MENU

                                2. EXIT

ENTER YOUR CHOICE :
```

OUTPUT FOR MAIN MENU



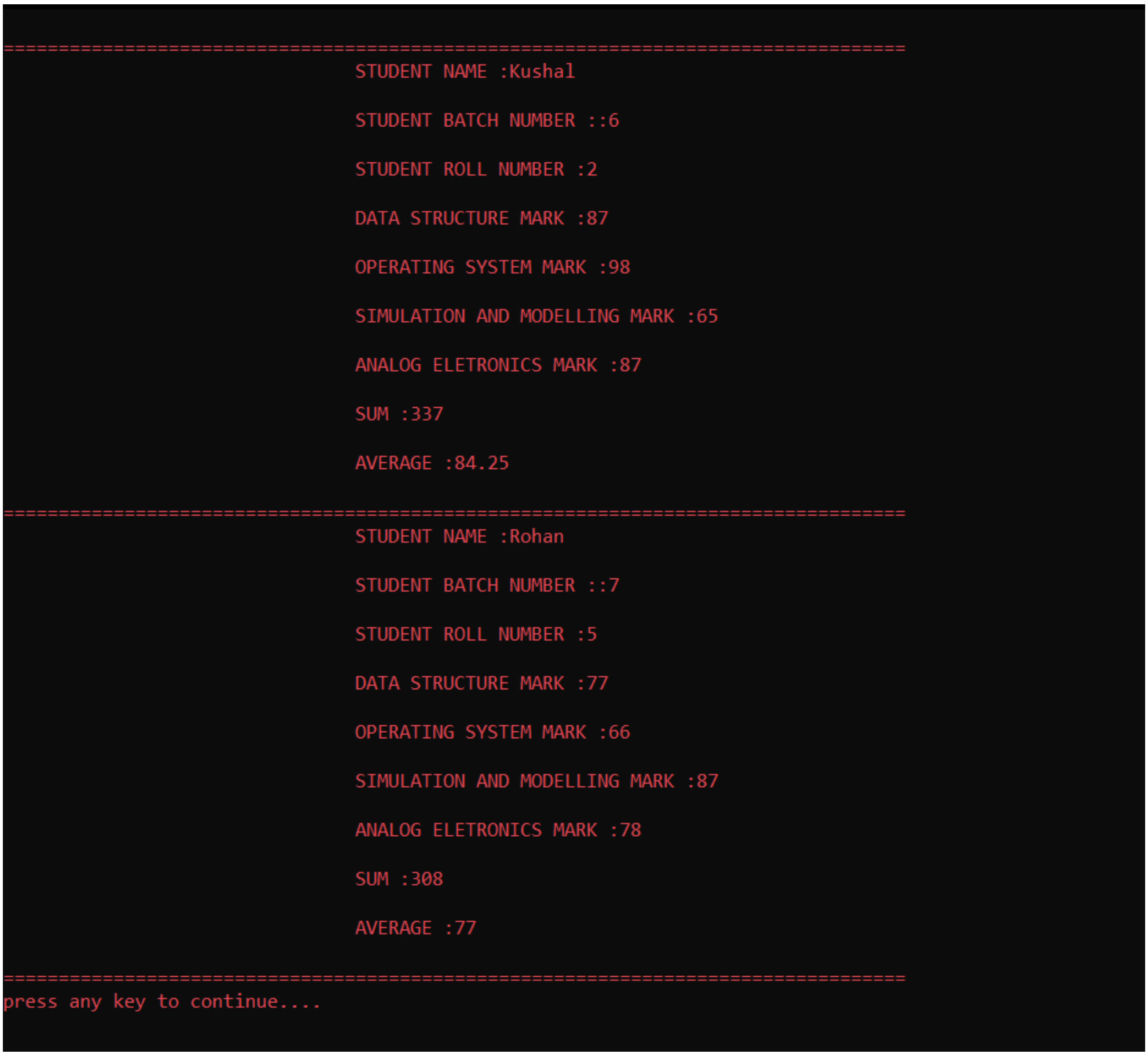
```
                                   =================MAIN MENU================


                                   1. CREATE STUDENT REPORT CARD


                                   2. VIEW ALL STUDENTS REPORT CARD


                                   3. VIEW A SINGLE STUDENT REPORT CARD


                                   4. MODIFY REPORT CARD


                                   5. RESULT


                                   6. DELETE RECORD


                                   ==============================
                                   ENTER YOUR CHOICE...:) <1-6> :
```

OUTPUT FOR CREATING STUDENT REPORT CARD



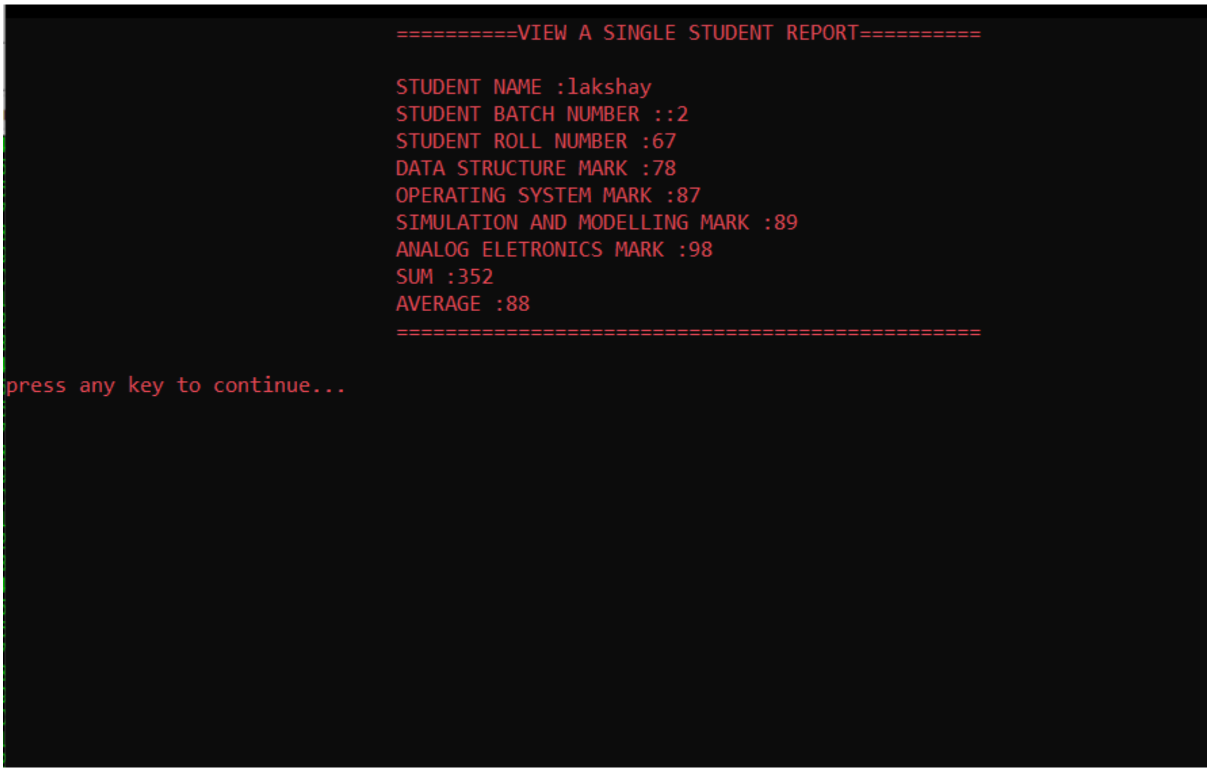
```
                                =======CREATE A REPORT CARD========

ENTER YOUR FULL NAME :Rohan
ENTER YOUR Rollno : 77
ENTER YOUR BATCH NUMBER :5
ENTER YOUR DATA STRUCTURE MARK :77
ENTER YOUR OPERATING SYSTEM MARK :66
ENTER YOUR SIMULATION AND MODELLING MARK :87
ENTER YOUR ANALOG ELETRONICS MARK :78

                                THE FILE IS SUCCESSFULLY SAVED

press any key to continue...
```

OUTPUT FOR DISPLAYING ALL RECORDS



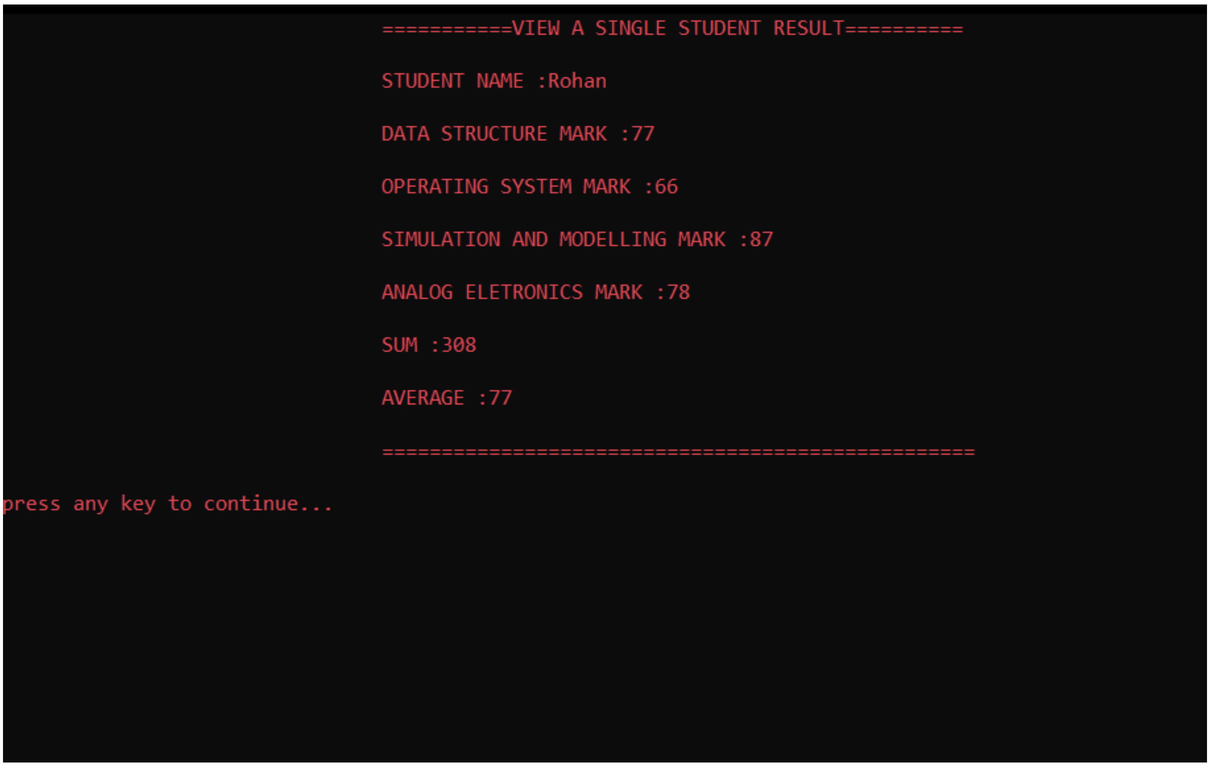
```
================================================================================
                                STUDENT NAME :Kushal

                                STUDENT BATCH NUMBER ::6

                                STUDENT ROLL NUMBER :2

                                DATA STRUCTURE MARK :87

                                OPERATING SYSTEM MARK :98

                                SIMULATION AND MODELLING MARK :65

                                ANALOG ELETRONICS MARK :87

                                SUM :337

                                AVERAGE :84.25

================================================================================
                                STUDENT NAME :Rohan

                                STUDENT BATCH NUMBER ::7

                                STUDENT ROLL NUMBER :5

                                DATA STRUCTURE MARK :77

                                OPERATING SYSTEM MARK :66

                                SIMULATION AND MODELLING MARK :87

                                ANALOG ELETRONICS MARK :78

                                SUM :308

                                AVERAGE :77

================================================================================
press any key to continue....
```

OUTPUT FOR DISPLAYING SPECIFIC RECORD



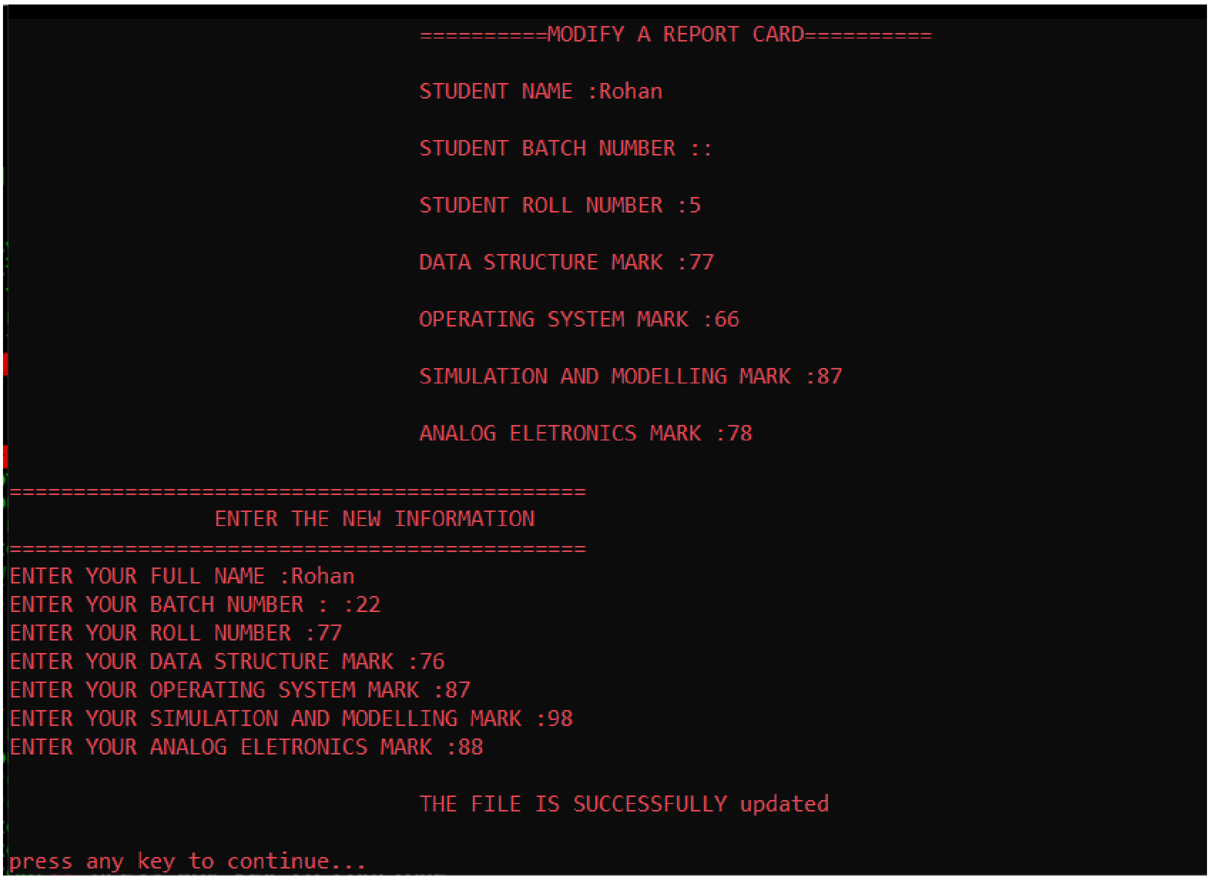
```
                                ==========VIEW A SINGLE STUDENT REPORT==========

                                STUDENT NAME :lakshay
                                STUDENT BATCH NUMBER ::2
                                STUDENT ROLL NUMBER :67
                                DATA STRUCTURE MARK :78
                                OPERATING SYSTEM MARK :87
                                SIMULATION AND MODELLING MARK :89
                                ANALOG ELETRONICS MARK :98
                                SUM :352
                                AVERAGE :88
                                ================================================

press any key to continue...
```

OUTPUT FOR RESULT OF THE STUDENT



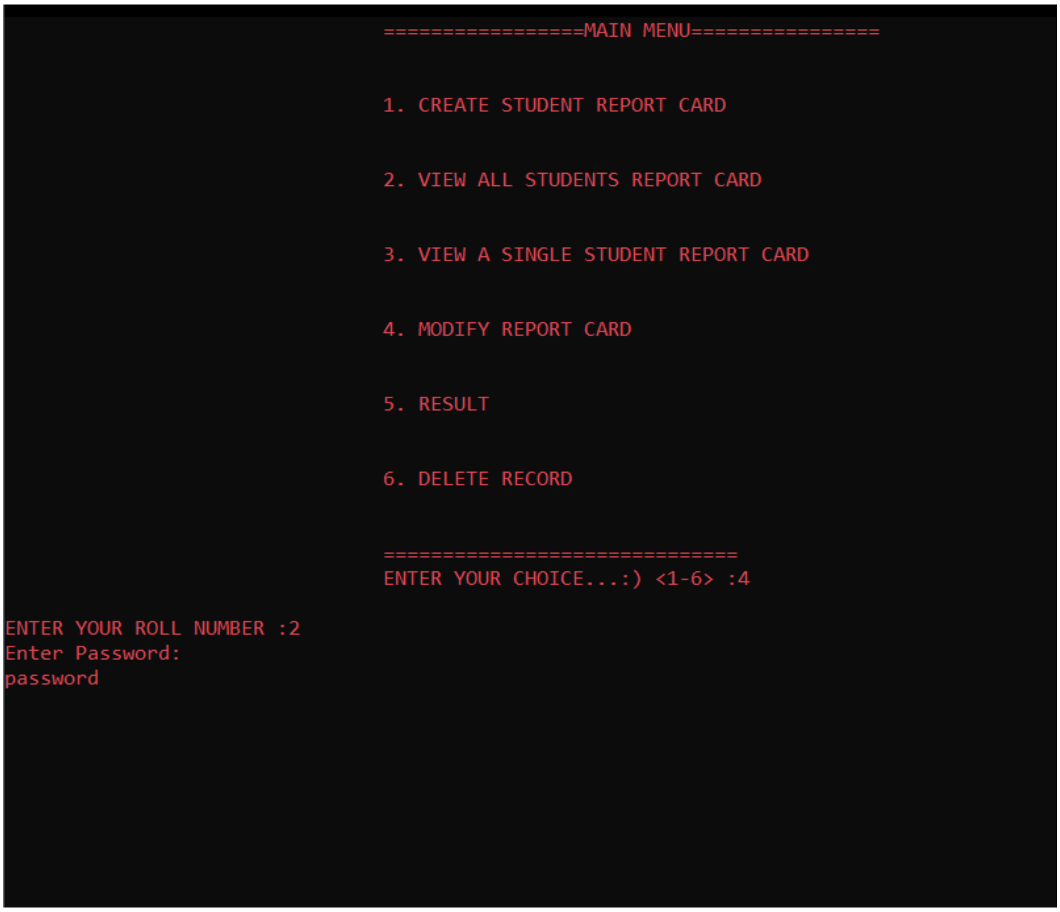
```
                                ===========VIEW A SINGLE STUDENT RESULT==========

                                STUDENT NAME :Rohan

                                DATA STRUCTURE MARK :77

                                OPERATING SYSTEM MARK :66

                                SIMULATION AND MODELLING MARK :87

                                ANALOG ELETRONICS MARK :78

                                SUM :308

                                AVERAGE :77

                                =================================================

press any key to continue...
```

OUTPUT FOR MODIFYING STUDENT RECORD



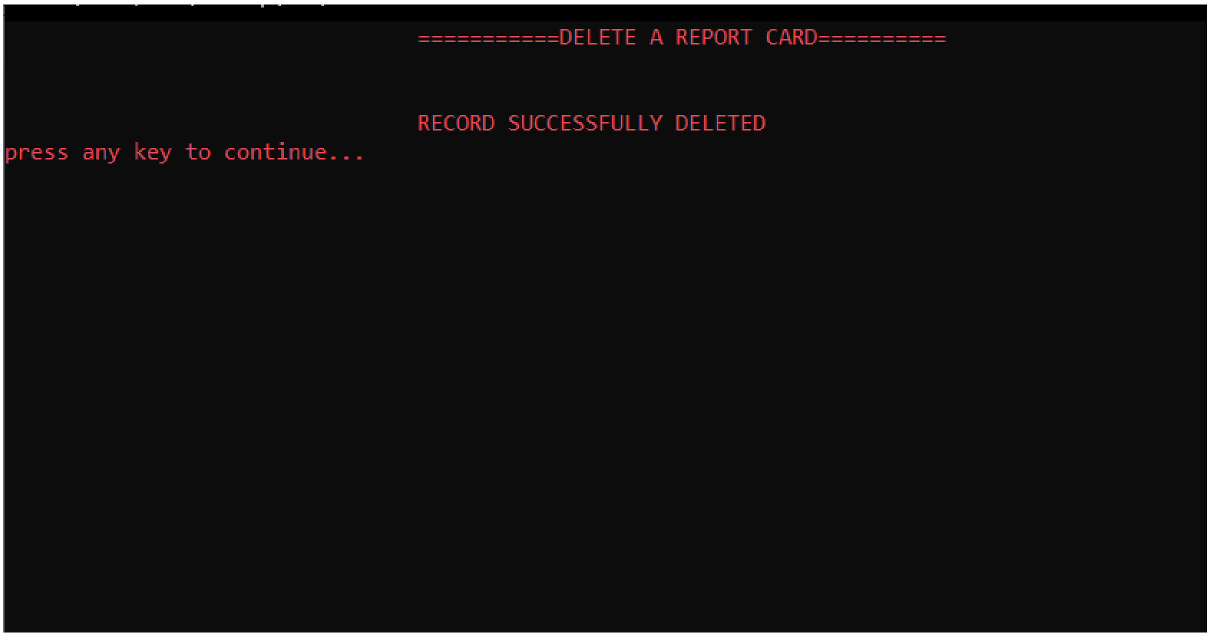
```
                                ==========MODIFY A REPORT CARD==========

                                STUDENT NAME :Rohan

                                STUDENT BATCH NUMBER ::

                                STUDENT ROLL NUMBER :5

                                DATA STRUCTURE MARK :77

                                OPERATING SYSTEM MARK :66

                                SIMULATION AND MODELLING MARK :87

                                ANALOG ELETRONICS MARK :78

=============================================
                ENTER THE NEW INFORMATION
=============================================
ENTER YOUR FULL NAME :Rohan
ENTER YOUR BATCH NUMBER : :22
ENTER YOUR ROLL NUMBER :77
ENTER YOUR DATA STRUCTURE MARK :76
ENTER YOUR OPERATING SYSTEM MARK :87
ENTER YOUR SIMULATION AND MODELLING MARK :98
ENTER YOUR ANALOG ELETRONICS MARK :88

                                THE FILE IS SUCCESSFULLY updated

press any key to continue...
```

PASSWORD FOR MODIFYING STUDENT RECORD



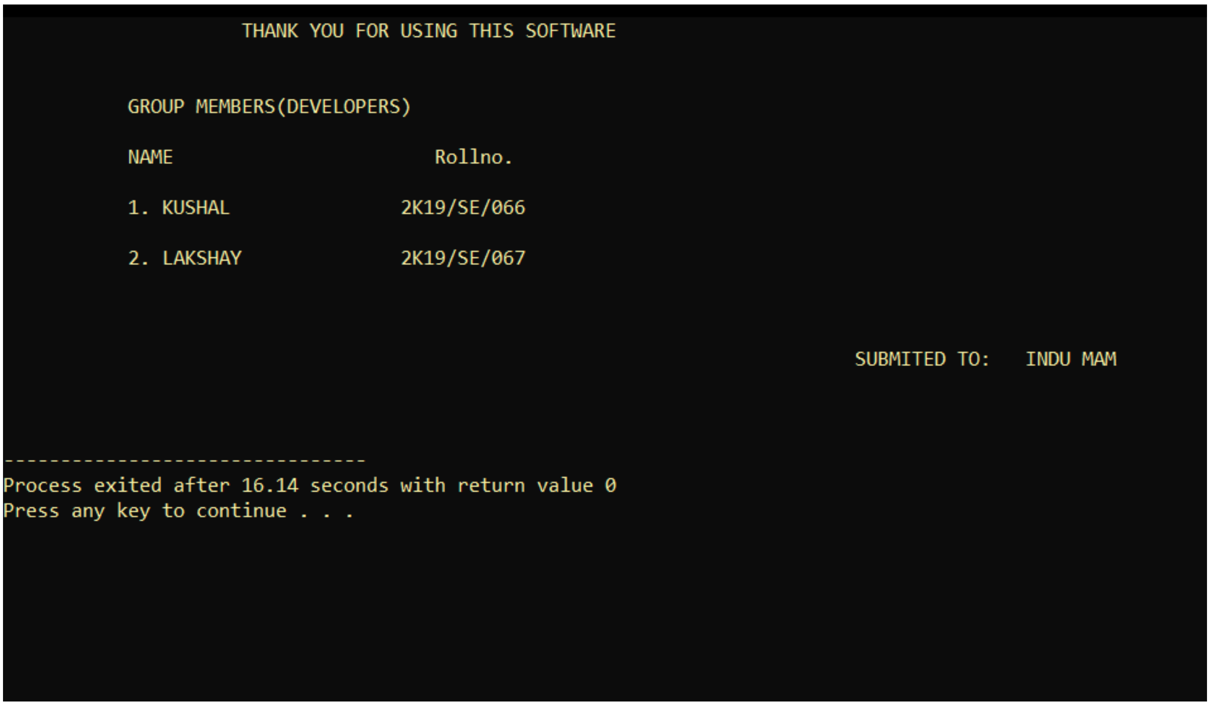
```
                                =================MAIN MENU================


                                1. CREATE STUDENT REPORT CARD


                                2. VIEW ALL STUDENTS REPORT CARD


                                3. VIEW A SINGLE STUDENT REPORT CARD


                                4. MODIFY REPORT CARD


                                5. RESULT


                                6. DELETE RECORD


                                ==============================
                                ENTER YOUR CHOICE...:) <1-6> :4

ENTER YOUR ROLL NUMBER :2
Enter Password:
password
```

OUTPUT FOR DELETING RECORD



```
                                ===========DELETE A REPORT CARD==========


                                RECORD SUCCESSFULLY DELETED
press any key to continue...
```

OUTPUT FOR EXITING SCREEN



```
                    THANK YOU FOR USING THIS SOFTWARE


          GROUP MEMBERS(DEVELOPERS)

          NAME                         Rollno.

          1. KUSHAL                 2K19/SE/066

          2. LAKSHAY                2K19/SE/067



                                                                          SUBMITED TO:    INDU MAM




--------------------------------
Process exited after 16.14 seconds with return value 0
Press any key to continue . . .
```

ADVANTAGES

- It saves the time of teacher to calculate the percentage, grade
- Makes easy to handle records
- Reduce the paper work
- Can easily change the records without cutting and any mess

DISADVANTAGES

- This is limited to small scale only
- This cannot be operated in networking over internet
- Password is visible

REFERENCES

- https://www.programiz.com/
- Computer Science With C++ by Sumita Arora
- Object-Oriented Programming With C++ by E. Balagurusamy
- https://stackoverflow.com