**STUDENT REPORT CARD**

**COMPUTER SCIENCE PROJECT**

Adhrika U Pai

INDEX

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **Slno** | **Topic** | **Page no.** | | 1. | Overview of C++ | 6 | | 2. | Need for the Project | 7 | | 3. | Hard ware and software requirements | 8 | | 4. | Flow chart or Data flow Diagram | 9 | | 5. | Header files used | 10 | | 6. | User defined functions/class/files | 11-19 | | 7. | Source code | 20-41 | | 8. | Output | 42-55 | | 9. | Short comings of the project | 56 | | 10. | Bibliography | 57 | |  |  |  | |

OVERVIEW OF C++

C++ is a statically typed, compiled, general-purpose, case-sensitive, free-form programming language that supports procedural, object-oriented, and generic programming. C++ is regarded as a middle-level language, as it comprises a combination of both high-level and low-level language features.

C++ was developed by Bjarne Stroustrup starting in 1979 at Bell Labs in Murray Hill, New Jersey, as an enhancement to the C language and originally named C with Classes but later it was renamed C++ in 1983. C++ is a superset of C, and that virtually any legal C program is a legal C++ program.

C++ is used by hundreds of thousands of programmers in essentially every application domain. C++ is being highly used to write device drivers and other software’s that rely on direct manipulation of hardware under real time constraints. C++ is widely used for teaching and research because it is clean enough for successful teaching of basic concepts

C++ supports a variety of programming styles. You can write in the style of Fortran, C, Smalltalk, etc., in any language. Each style can achieve its aims effectively while maintaining runtime and space efficiency.

The most important thing to do when learning C++ is to focus on concepts and not get lost in language technical details. The purpose of learning a programming language is to become a better programmer; that is, to become more effective at designing and implementing new systems and at maintaining old ones.

NEED FOR THE PROJECT

A report card communicates a student's performance academically. Report cards are now frequently issued in automated form by computers and may be mailed to parents and students. A typical report card uses a grading scale to determine the quality of a student's school work. The Report Cards are produced to provide users comprehensive information on all the vital parameters, be it student, teacher or school related variables, yet concise, accurate information about each school in a standard format which is easy to understand and allows meaningful comparisons to be made among schools.

This C++ project on STUDENT REPORT CARD has student class with data members like admnno,roll no, name, marks and grade. Member functions in this class are used for accept / display academical details of students, a function to calculate grade based on marks obtained by student, function to display results in tabular form, function to return rollno and function to return percentage. Student Records are stored in binary file.

It also has admission class with data members like admnno, f\_name, m\_name, f\_no, m\_no, hno, hname, landmark, city, state. Member functions in this class are used for accept / display admission details of students and a function to return admission number. Admission Records are stored in another binary file.

This menu driven program illustrates read, write, search, modify and delete operations in both the binary files. This program provides the details and marks of students.

This program contains several functions to create /display records in binary files, display particular record, display distinctions and failures, modify a record and delete a record in the academical section. In administrative section we have functions to create/display records, display particular record, modify a record, delete a record .

Hard ware and software requirements

The platform used is C++. Hence we decided to use Microsoft Word 2016.

Henceforth for optimal usage of such software a windows based operating system preferably Windows 10 must be there. Also on the hardware part any system having Windows 10 installed will suffice.

Flow chart or Data flow Diagram

HEADER FILES USED

* fstream.h
* string.h
* iomanip.h
* stdio.h
* conio.h
* stdlib.h

USER DEFINED FUNCTIONS/CLASS/FILES

1. An object ad of **class admission** is declared with following members:

a. private: admnno, f\_name, m\_name, f\_no, m\_no, hno, hname, landmark, city, state

b. public: void get() to read the details of the student

void put() to display details of students

long int retadmno() to return the admission number

2. An object st of **class student** is declared with following members:

a. private: admnno, rollno, name, p\_marks, c\_marks, m\_marks, e\_marks, cs\_marks, per, grade

void calculate() to calculate grade according to following instructions:

per=(p\_marks+c\_marks+m\_marks+e\_marks+cs\_marks)/5

percentage grade

>=80 A

>=60 B

>=50 C

>=40 D

<40 F

b. public: void getdata() to enter academical results of the student

void showdata() to display academical results of the student

void show\_tabular() to display academical results of student in a table

int retrollno() to return rollno

float rettotal() to return per

3. FUNCTION 1(**write\_student()**)

a. Start

b. A binary file named "Results " is opened in ofstream and appending mode

c. st.getdata() is called.

d. This is written in the file

e. File closed

f. Stop

4. FUNCTION 2(**display\_all()**)

a. Start

b. Binary file named "Results" is opened in ifstream.

c. if !fin then

d. print "File could not be opened"

e. while (fin.read((char\*)&st,sizeof(st)))

f. st.showdata() is called.

g. File closed

h. Stop

5. FUNCTION 3(**display\_sp(n)**)

a. Start

b. Binary file named “Results” is opened in ifstream.

c. if !fin then

d. print "File could not be opened"

e. while (fin.read((char\*)&st,sizeof(st)))

{

f. if st.retrollno=n

st.showdata is called and set flag=1

}

g. File closed

h. if file=0 then

print “Record does not exist"

i. Stop

6. FUNCTION 4(**write\_admn()**)

a. Start

b. A binary file named "news" is opened in ofstream and appending mode

c. ad.get() is called.

d. This is written in the file

e. File closed

f. Stop

7. FUNCTION 5(**display\_spad(n)**)

a. Start

b. Binary file named"news"is opened in ifstream.

c. if !fin then

d. print "File could not be opened"

e. while (fin.read((char\*)&ad,sizeof(ad))

{

f. if ad.admnno=n

st.put is called and set flag=1

}

g. File closed

h. if file=0 then

print"record does not exist"

i. Stop

8. FUNCTION 6(**copy()**)

a. Start

b. A binary file named "Results" is opened in ifstream

c. A binary file named "Failuress" is opened in ofstream mode

d. while(fin.read((char\*)&st,sizeof(st)))

e. if st.rettotal()<40

f. write it into "Failuress" file

g. Close both files

h. Stop

9. FUNCTION 7(**dispfile()**)

a. Start

b. Binary file named "Failures.dat" is opened in ifstream.

c. st.showdata() is called.

d. File closed

e. Stop

10. FUNCTION 8(**copy1()**)

a. Start

b. A binary file named "Results" is opened in ifstream

c. A binary file named "Distinctionss" is opende in ofstream mode

d. while(fin.read((char\*)&st,sizeof(st)))

e. if st.rettotal()>70

f. write it into "Distinctionssa" file

g. Close both files

h. Stop

11. FUNCTION 9(**dispfile1()**)

a. Start

b. Binary file named "Distinctionss" is opened in ifstream.

c. st.showdata() is called.

d. File closed

e. Stop

12. FUNCTION 10(**modify\_student(n)**)

a. Start

b. A binary file "Results" is opened in fstream with ios::in,ios::out as file modes.

c. if !file then

d. print"File could not be opened"

e. while (file.read((char\*)&st,sizeof(st))) repeat steps f to l

f. if st.retrollno()=n

{

g. st.showdata() is called.

h. Read new information from user.

i. Set pos=(-1)\*sizeof(st)

j. Place seek pointer(pos,ios::curr)

k. Write to the file.

l. Set found=1

}

m. Close the file

n. if found =0 then

o. print "Record not found"

p. Stop

13. FUNCTION 11(**delete\_student()**)

a. Start

b. A binary file named "Results" in ifstream.

c. if !fin then

d. print"File could not be opened"

e. A binary file named "Temps" in ofstream

f. Set flag=0

g. while(fin.read((char\*)&st,sizeof(st)))

if st.rerollno!=n

write to the "temps"file

h. if flag=0

print"Record not found"

i. else

j. print "Record deleted"

k. remove "Results"

l. rename "temps" to "Results"

m. Close both files

n. Stop

14. FUNCTION 12(**modify\_ad(n)**)

a. Start

b. A binary file "news" is opened in fstream with ios::in,ios::out as file modes.

c. if !file then

d. print "File could not be opened"

e. while (file.read((char\*)&ad,sizeof(ad))) repeat steps f to l

f. if ad.retadmno()=n

{

g. ad.put() is called.

h. Read new information from user.

i. Set pos=(-1)\*sizeof(st)

j. Place seek pointer(pos,ios::curr)

k. Write to the file.

l. Set found=1

}

m. Close the file

n. if found =0 then

o. print "Record not found"

p. Stop

15. FUNCTION 13(**delete\_ad()**)

a. Start

b. A binary file named "news" in ifstream.

c. if !fin then

d. print "File could not be opened"

e. A binary file named "Temps" in ofstream

f. Set flag=0

g. while(fin.read((char\*)&ad,sizeof(ad)))

if ad.retadmno!=n

write to the "temps"file

h. if flag=0 then

print "Record not found"

i. else

{

j. print "Record deleted"

k. remove "Results"

l. rename "temps" to "Results"

}

m. Close both files

n. Stop

16. FUNCTION 14(**class\_result()**)

a. Start

b. A binary file named "news" in ifstream.

c. if !fin then

d. print "File could not be opened"

e. while(fin.read((char\*)&st,sizeof(st)))

st.show\_tabular();

f. Close both files

g. Stop

17. FUNCTION 15(**result()**)

a. Start.

b. Declare ch, rno.

c. Display menu

d. Read choice from user.

e. if choice is 1:class\_result() is called

f. if choice is 2: Read roll number of student

display\_sp(rno) is called

g. if choice is 3:copy() and dispfile() is called

h. if choice is 4:copy1() and dispfile1() is called

i. if choice is 5: exit program

j. Stop

18. FUNCTION 16(**entry\_menu()**)

a. Start

b. Declare int ch, num

c. Display Menu

d. Read the choice from user

e. if choice is 1:write\_student is called

f. if choice is 2:display\_all() is called

g. if choice is 3:Read the rollnumber of the student whose record to be displayed

display\_sp(num) is called

h. if choice is 4:Read the rollnumber of student whose record to be modified

modify\_student(num) is called

i. if choice is 5:Read the rollnumber of student whose record to be deleted

delete\_student(num) is called

j. if choice is 6 exit the program

k. Stop

19. FUNCTION 17(**admnres()**)

a. Start

b. Declare int ch,num

c. Display Menu

d. Read the choice from user

e. if choice is 1:write\_admn is called

f. if choice is 2:Read the rollnumber of the student whose record to be displayed

display\_spad(num) is called

g. if choice is 3:Read the rollnumber of student whose record to be modified

modify\_ad(num) is called

h. if choice is 4:Read the rollnumber of student whose record to be deleted

delete\_ad(num) is called

i. if choice is 5 exit the program

j. Stop

20. FUNCTION 18(**main()**)

a. Start

b. Declare int ch

c. Display Menu

d. Read choice from user

e. if choice is 1: result() is called

f. if choice is 2: entry\_menu() is called

g. if choice is 3: amdmnres() is called

h. if choice is 4: exit the program

i.Stop

SOURCE CODE

*//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*// HEADER FILE USED IN PROJECT*

*//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

#include<fstream.h>

#include<string.h>

#include<iomanip.h>

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//                   *CLASSES  USED IN PROJECT*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class admission

{

private:

long int admnno;

char f\_name[50];

char m\_name[80];

char f\_no[11];

char m\_no[11];

int hno;

char hname[50];

char landmark[50];

char city[50];

char state[50];

public:

void get()

{

cout<<"\nAdmission number: ";

cin>>admnno;

cout<<"\nFather's Name: ";

gets(f\_name);

cout<<"\nMother's Name: ";

gets(m\_name);

cout<<"\nFather's Phone Number: ";

gets(f\_no);

cout<<"\nMother's Phone Number: ";

gets(m\_no);

cout<<"\nAddress:\n";

cin>>hno;

gets(hname);

gets(landmark);

gets(city);

gets(state);

}

void put()

{

cout<<"\nAdmission Number: "<<admnno

     <<"\nFather's Name: "<<f\_name

     <<"\nMother's Name: "<<m\_name

     <<"\nFather's Phone Number: "<<f\_no

     <<"\nMother's Phone Number: "<<m\_no

    <<"\nAddress: \n"

     <<hno<<"."<<hname<<"\n"<<landmark

     <<"\n"<<city<<"\n"<<state;

}

long int retadmno()

{

return admnno;

}

}ad;

class student

{

private:

long int admnno;

int rollno;

char name[50];

float p\_marks,c\_marks,m\_marks,e\_marks,cs\_marks;

float per;

char grade;

void calculate()

{

per=(p\_marks+c\_marks+m\_marks+e\_marks+cs\_marks)/5;

if(per>=80)

grade='A';

else

if(per>=60)

grade='B';

else

if(per>=50)

grade='C';

else

if(per>=40)

grade='D';

else

grade='F';

}

public:

void getdata()

{

cout<<"\nEnter the Admission number of student ";

cin>>admnno;

cout<<"\nEnter the Roll number of student ";

cin>>rollno;

cout<<"\nEnter the Name of student ";

gets(name);

cout<<"\nEnter the Marks in PHYSICS out of 100: ";

cin>>p\_marks;

cout<<"\nEnter the Marks in CHEMISTRY out of 100: ";

cin>>c\_marks;

cout<<"\nEnter the Marks in MATHS out of 100: ";

cin>>m\_marks;

cout<<"\nEnter the Marks in ENGLISH out of 100: ";

cin>>e\_marks;

cout<<"\nEnter the Marks in COMPUTER SCIENCE out of 100: ";

cin>>cs\_marks;

calculate();

}

void showdata()

{

cout<<"\nRoll number of student: "<<rollno

     <<"\nName of student: "<<name

     <<"\nMarks in PHYSICS: "<<p\_marks

     <<"\nMarks in CHEMISTRY: "<<c\_marks

     <<"\nMarks in MATHS: "<<m\_marks

     <<"\nMarks in ENGLISH: "<<e\_marks

     <<"\nMarks in COMPUTER SCIENCE: "<<cs\_marks

     <<"\nPERCENTAGE of student: "<<per

    <<"\nGRADE of student: "<<grade;

}

void show\_tabular()

{

cout<<rollno<<setw(6)<<" "<<name<<setw(5)<<"\t"<<p\_marks

     <<setw(4)<<c\_marks<<setw(4)<<m\_marks<<setw(4)

     <<e\_marks<<setw(4)<<cs\_marks<<setw(4)<<"  "<<per<<"\t"<<"\t"

     <<grade<<endl;

}

int retrollno()

{

return rollno;

}

float rettotal()

{

return per;

}

}st;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to write in file*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void write\_student()

{

ofstream fout("Results",ios::binary|ios::app);

st.getdata();

fout.write((char\*)&st,sizeof(student));

fout.close();

cout<<"\n\nSTUDENT RECORD has been created";

cin.ignore();

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to display all students list*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void display\_all()

{

ifstream fin;

fin.open("Results",ios::binary);

if(!fin)

{

cout<<"File could not be opened!!"<<"\nPress any key...";

getch();

return;

}

cout<<"\n\n\n\t\t\t\t\tDISPLAY ALL RECORD!!!\n\n";

while(fin.read((char\*)&st,sizeof(student)))

{

st.showdata();

cout<<"\n\n=======================================================\n";

}

fin.close();

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to read specific record from file*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void display\_sp(int n)

{

ifstream fin;

fin.open("Results",ios::binary);

if(!fin)

{

cout<<"File could not be opened!!"<<"\nPress any key...";

getch();

return;

}

int flag=0;

while(fin.read((char\*)&st,sizeof(student)))

{

if(st.retrollno()==n)

{

st.showdata();

flag=1;

}

}

fin.close();

if(flag==0)

cout<<"\n\nRECORD DOESN'T EXIST";

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to write in file*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void write\_admn()

{

ofstream fout("news",ios::binary|ios::app);

ad.get();

fout.write((char\*)&ad,sizeof(ad));

fout.close();

cout<<"\n\nSTUDENT RECORD has been created";

cin.ignore();

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to read specific record from file*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void display\_spad(int n)

{

ifstream fin;

fin.open("news",ios::binary);

if(!fin)

{

cout<<"File could not be opened!!"<<"Press any key...";

getch();

return;

}

int flag=0;

while(fin.read((char\*)&ad,sizeof(ad)))

{

if(ad.retadmno()==n)

{

ad.put();

flag=1;

}

}

fin.close();

if(flag==0)

cout<<"\n\nRECORD DOESN'T EXIST";

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// *function to create a file with all failures*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void copy()

{

ifstream fin("Results",ios::binary);

ofstream fout("Failuress",ios::binary);

while(fin.read((char\*)&st,sizeof(st)))

{

if(st.rettotal()<40)

fout.write((char\*)&st,sizeof(st));

}

fin.close();

fout.close();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to display all failures*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void dispfile()

{

ifstream fin("Failuress",ios::binary);

while(fin.read((char\*)&st,sizeof(st)))

st.showdata();

fin.close();

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to create a file with all distinctions*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void copy1()

{

ifstream fin("Results",ios::binary);

ofstream fout("Distinctionss",ios::binary);

while(fin.read((char\*)&st,sizeof(st)))

{

if(st.rettotal()>70)

fout.write((char\*)&st,sizeof(st));

}

fin.close();

fout.close();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to display all distinctions*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void dispfile1()

{

ifstream fin("Distinctionss",ios::binary);

while(fin.read((char\*)&st,sizeof(st)))

st.showdata();

fin.close();

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to modify record of file*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void modify\_student(int n)

{

int found=0;

fstream file;

file.open("Results",ios::binary|ios::in|ios::out);

if(!file)

{

cout<<"File could not be opened!!"<<"\nPress any key...";

getch();

return;

}

while(file.read((char\*)&st,sizeof(student))&&found==0)

{

if(st.retrollno()==n)

{

st.showdata();

cout<<"\n\nPlease enter the NEW DETAILS of student"<<endl;

st.getdata();

int pos=(-1)\*sizeof(st);

file.seekp(pos,ios::cur);

file.write((char\*)&st,sizeof(student));

cout<<"\n\n\tRECORD UPDATED";

found=1;

}

}

file.close();

if(found==0)

cout<<"\n\nRECORD NOT FOUND";

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to delete record of file*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void delete\_student(int n)

{

ifstream fin;

fin.open("Results",ios::binary);

if(!fin)

{

cout<<"File could not be opened!!"<<"\nPress any key...";

getch();

return;

}

ofstream fout("Temps",ios::binary);

int flag=0;

while(fin.read((char\*)&st,sizeof(student)))

{

if(st.retrollno()!=n)

{

fout.write((char\*)&st,sizeof(student));

flag=1;

}

}

if(flag==0)

cout<<"RECORD NOT FOUND";

else

{

cout<<"\n\n\tRecord deleted";

remove("Results");

rename("Temps","Results");

}

fin.close();

fout.close();

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to modify record of file*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void modify\_ad(int n)

{

int found=0;

fstream file;

file.open("news",ios::binary|ios::in|ios::out);

if(!file)

{

cout<<"File could not be opened!!"<<"\nPress any key...";

getch();

return;

}

while(file.read((char\*)&ad,sizeof(ad))&&found==0)

{

if(ad.retadmno()==n)

{

ad.put();

cout<<"\n\nPlease enter the NEW DETAILS of student"<<endl;

ad.get();

int pos=(-1)\*sizeof(ad);

file.seekp(pos,ios::cur);

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

cout<<"\n\n\tRECORD UPDATED";

found=1;

}

}

file.close();

if(found==0)

cout<<"\n\nRECORD NOT FOUND";

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to delete record of file*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void delete\_ad(int n)

{

ifstream fin;

fin.open("news",ios::binary|ios::in|ios::out);

if(!fin)

{

cout<<"File could not be opened!!"<<"\nPress any key...";

getch();

return;

}

ofstream fout("Temps",ios::binary);

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

int flag=0;

while(fin.read((char\*)&ad,sizeof(ad)))

{

if(ad.retadmno()!=n)

{

fout.write((char\*)&ad,sizeof(ad));

flag=1;

}

}

if(flag==0)

cout<<"RECORD NOT FOUND";

else

{

cout<<"\n\n\tRecord deleted";

remove("Results");

rename("Temps","Results");

}

fin.close();

fout.close();

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to display all students grade report*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void class\_result()

{

ifstream fin;

fin.open("Results",ios::binary);

if(!fin)

{

cout<<"File could not be opened!!"<<"\nPress any key...";

getch();

return;

}

cout<<"\n\n\t\t\t\tALL STUDENTS RESULT\n\n";

cout<<"================================================================================";

cout<<"\nR.NO   NAME             P   C   M   E   CS   PERCENTAGE      GRADE"<<endl;

while(fin.read((char\*)&st,sizeof(student)))

{

st.show\_tabular();

}

fin.close();

getch();

}

 //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *function to display result menu*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void result()

{

int ch;

int rno;

cout<<"\n\n\n\t\t\t\tRESULT MENU"

<<"\n================================================================================"

     <<"\n\n\t\t\t\t1.CLASS REPORT"

     <<"\n\n\t\t\t\t2.STUDENT REPORT CARD"

     <<"\n\n\t\t\t\t3.FAILURES"

     <<"\n\n\t\t\t\t4.DISTINCTIONS"

     <<"\n\n\t\t\t\t5.BACK TO MAIN MENU"

     <<"\n\n\n\t\t\tEnter your choice: ";

cin>>ch;

clrscr();

switch (ch)

{

case 1:class\_result();

break;

case 2:cout<<"\n\n\tEnter the Roll number of student: ";

cin>>rno;

display\_sp(rno);

break;

case 3:copy();

dispfile();

break;

case 4:copy1();

dispfile1();

break;

case 5:break;

default:cout<<"\nWRONG ENTRY";

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *ENTRY / EDIT MENU FUNCTION*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void entry\_menu()

{

int ch,num;

clrscr();

cout<<"\n\n\n\t\t\t\tENTRY MENU"

<<"\n================================================================================"

     <<"\n\n\t\t\t1.CREATE STUDENT RECORD"

     <<"\n\n\t\t\t2.DISPLAY ALL STUDENTS RECORDS"

     <<"\n\n\t\t\t3.SEARCH STUDENT RECORD"

     <<"\n\n\t\t\t4.MODIFY STUDENT RECORD"

    <<"\n\n\t\t\t5.DELETE STUDENT RECORD"

    <<"\n\n\t\t\t6.BACK TO MAIN MENU"

     <<"\n\n\n\t\tPlease enter your choice: ";

cin>>ch;

clrscr();

switch(ch)

{

case 1:write\_student();

break;

case 2:display\_all();

break;

case 3:cout<<"\n\n\tPlease enter Roll number";

cin>>num;

display\_sp(num);

break;

case 4: cout<<"\n\n\tPlease enter Roll number";

cin>>num;

modify\_student(num);

break;

case 5:cout<<"\n\n\tPlease enter Roll number";

cin>>num;

delete\_student(num);

break;

case 6:break;

default:cout<<"\nWRONG ENTRY";entry\_menu();

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//     *ADMINISTRATOR MENU FUNCTION*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void admnres()

{

int ch,num;

clrscr();

cout<<"\n\n\n\t\t\t\tADMISSION DETAILS"

<<"\n================================================================================"

     <<"\n\n\t\t\t1.ENTER ADMISSION DETAILS OF STUDENT "

    <<"\n\n\t\t\t2.SEARCH FOR ADMISSION DETAILS"

    <<"\n\n\t\t\t3.MODIFY ADMISSION DETAILS"

     <<"\n\n\t\t\t4.DELETE ADMISSION DETAILS"

    <<"\n\n\t\t\t5.BACK TO MAIN MENU"

    <<"\n\n\n\t\tPlease enter your choice: ";

cin>>ch;

clrscr();

switch(ch)

{

case 1:write\_admn();

break;

case 2:cout<<"\n\n\tPlease enter Admission number";

cin>>num;

display\_spad(num);

break;

case 3: cout<<"\n\n\tPlease enter Admission number";

cin>>num;

modify\_ad(num);

break;

case 4:cout<<"\n\n\tPlease enter Admission number";

cin>>num;

delete\_ad(num);

break;

case 5:break;

default:cout<<"\nWRONG ENTRY";entry\_menu();

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//   *THE MAIN FUNCTION OF PROGRAM*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void main()

{

int ch;

clrscr();

textcolor(GREEN);

textbackground(DARKGRAY);

do

{

clrscr();

cout<<"\n\n\n\t\t\t\t";

     textcolor(LIGHTRED);

     cprintf("MAIN MENU");

     textcolor(YELLOW);

     cout<<"\n";

cprintf("================================================================================");

   cout<<"\n\n\t\t\t\t1.RESULT MENU"

     <<"\n\n\t\t\t\t2.ENTRY/EDIT MENU"

    <<"\n\n\t\t\t\t3.ADMISSION DETAILS"

    <<"\n\n\t\t\t\t4.EXIT"

     <<"\n\n\n\t\t\t";

    textcolor(CYAN);

    cprintf("Please select your option: ");

cin>>ch;

clrscr();

switch(ch)

{

case 1:result();

break;

case 2:entry\_menu();

break;

case 3:admnres();

break;

case 4:break;

default:cout<<"\WRONG ENTRY";

}

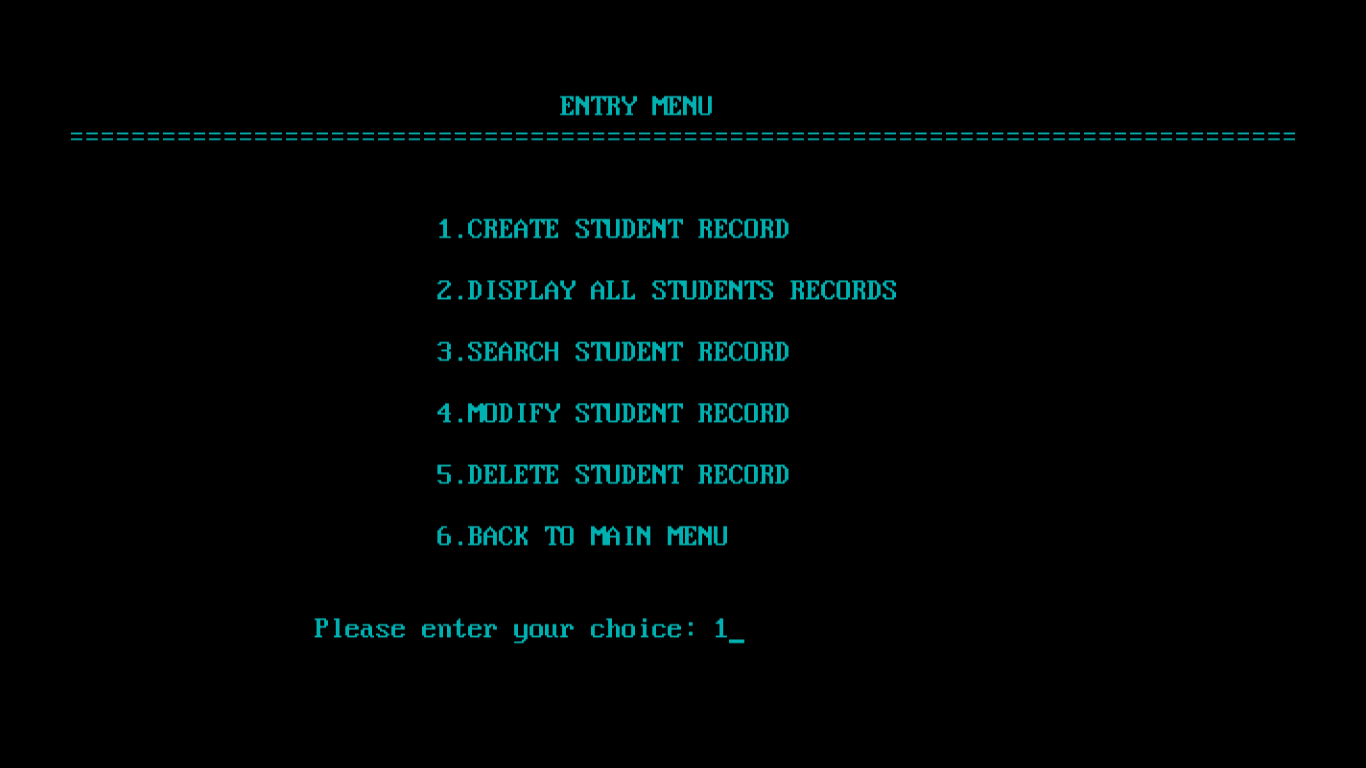
}while(ch!=4);

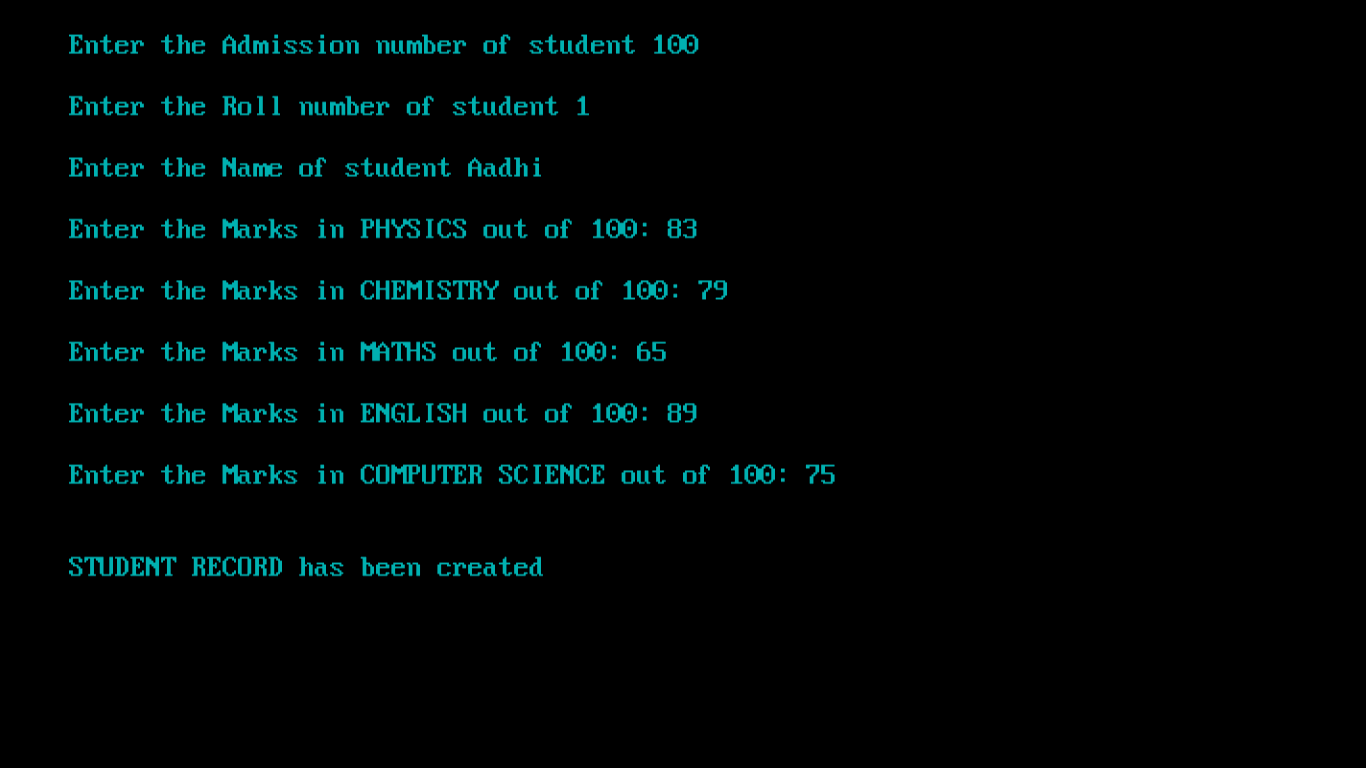
}

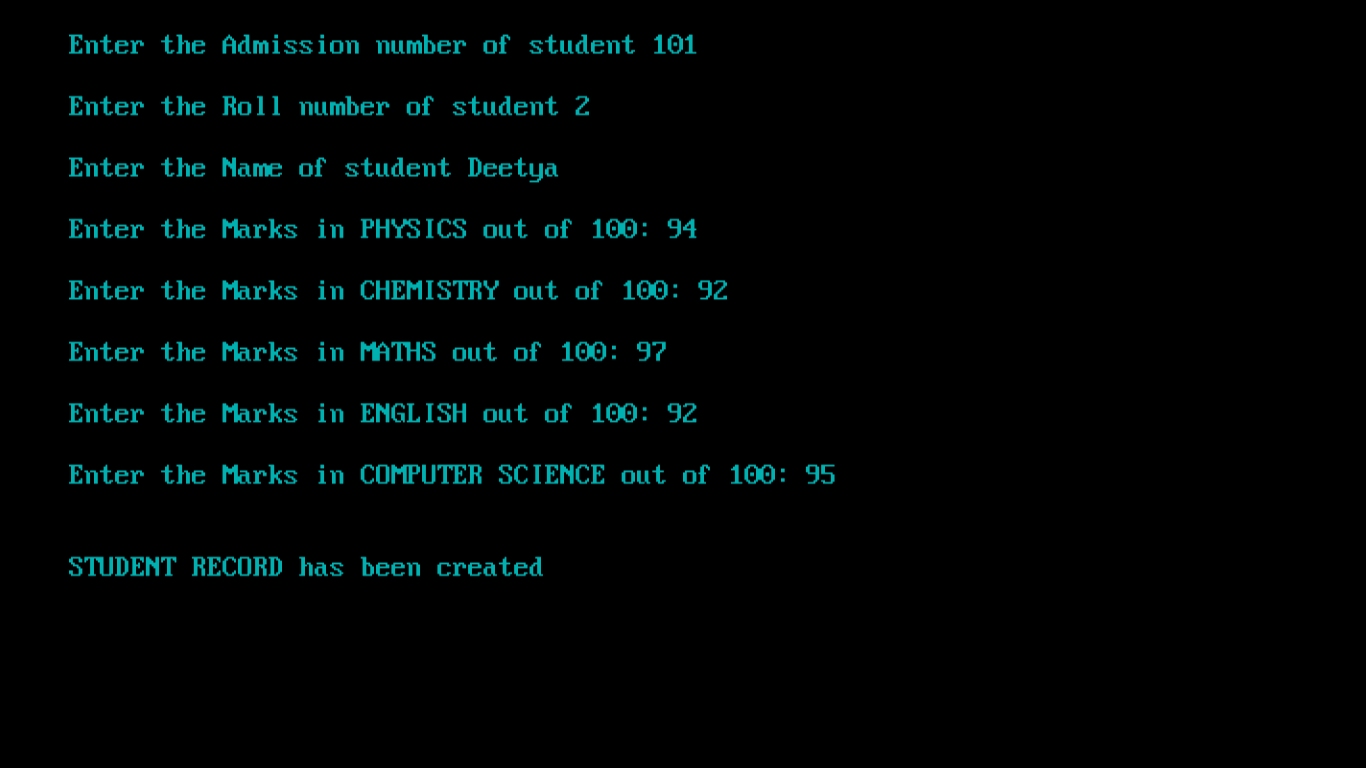
OUTPUT

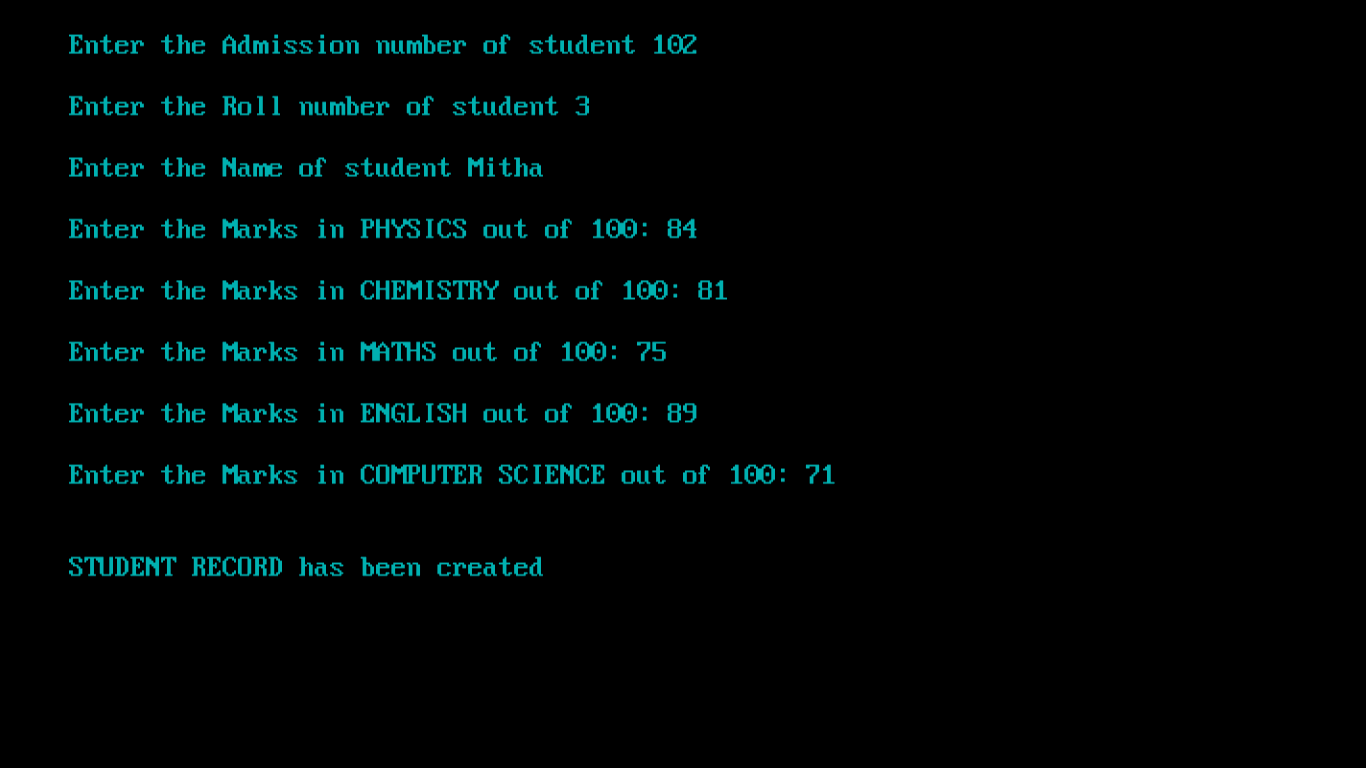


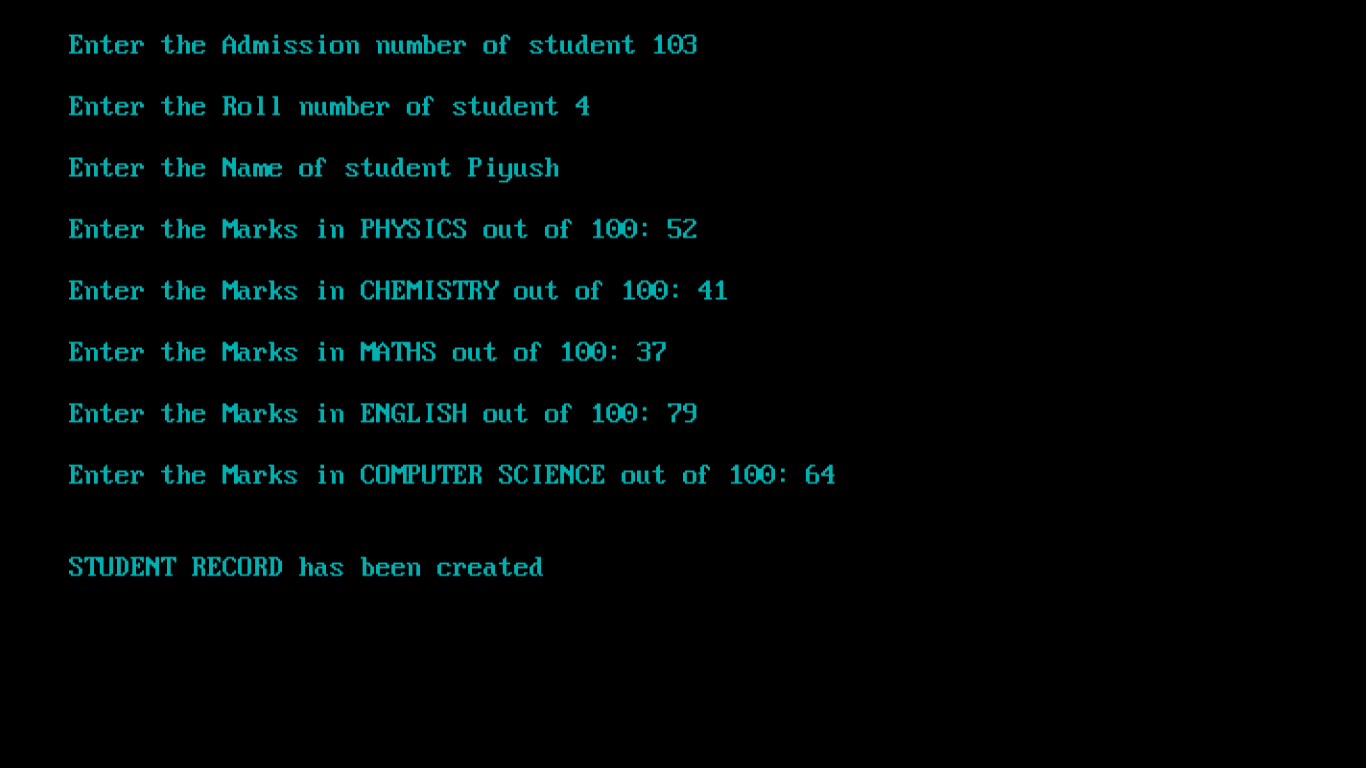


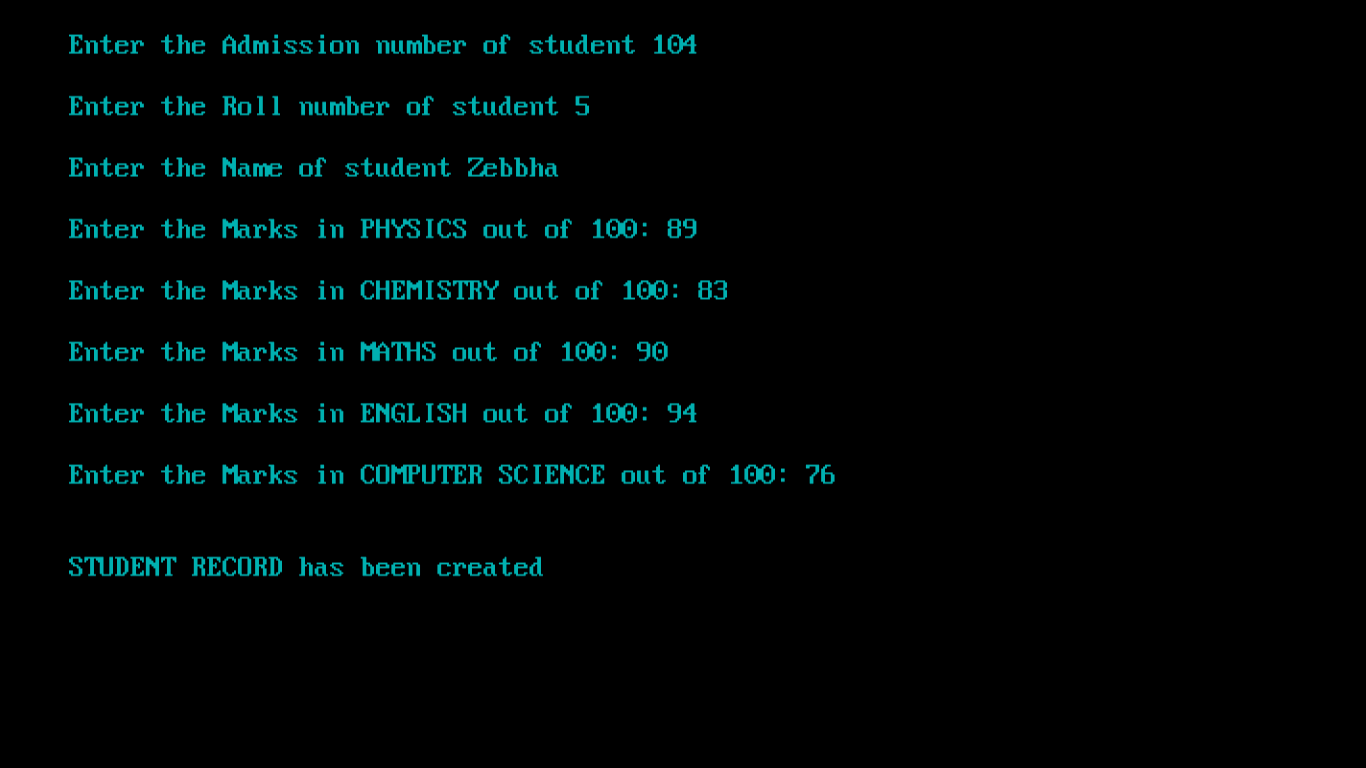


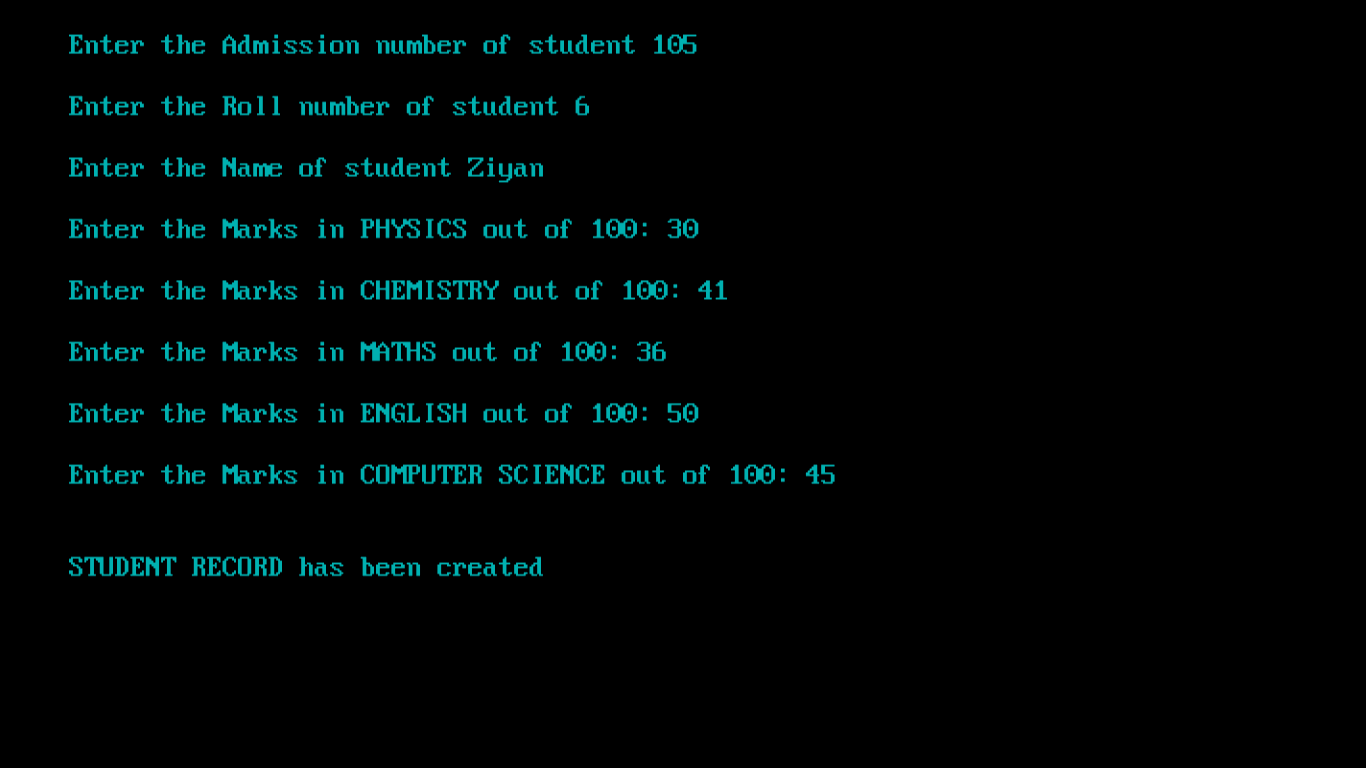






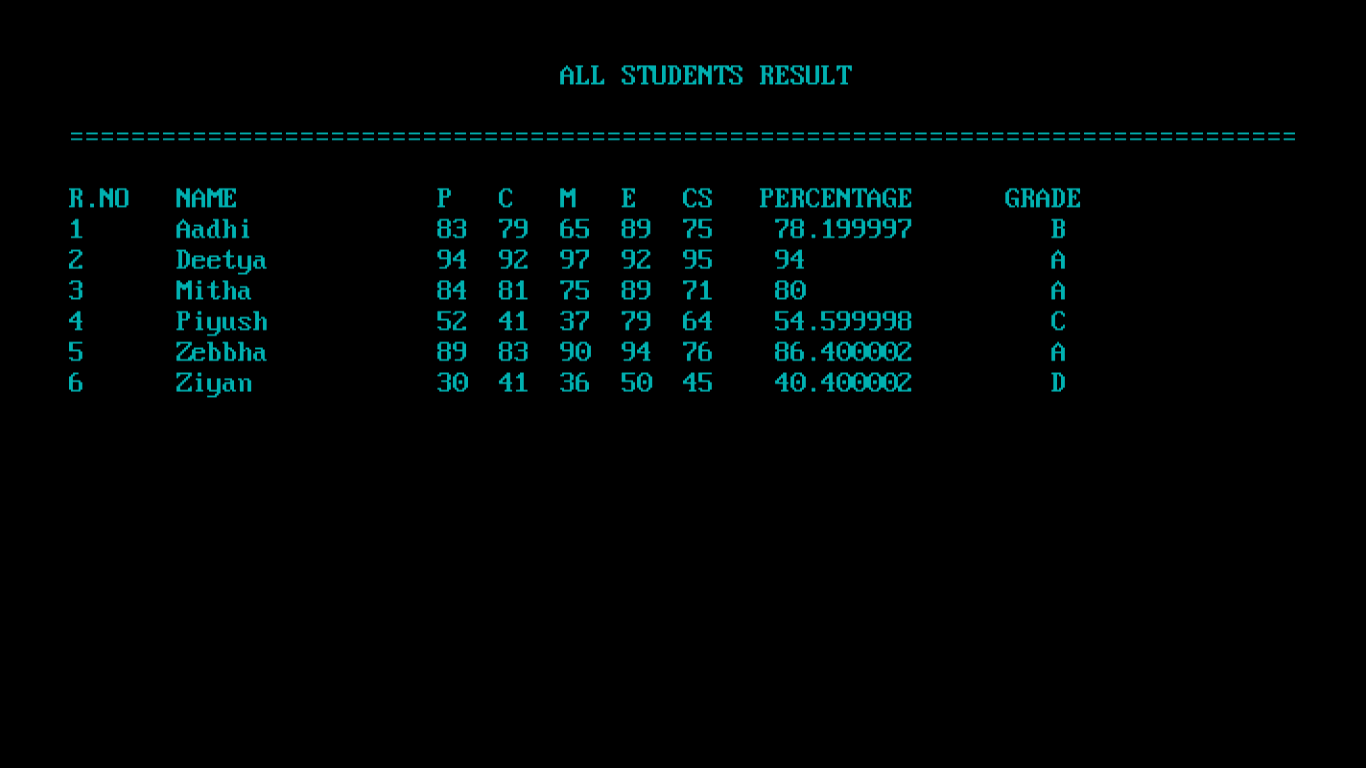






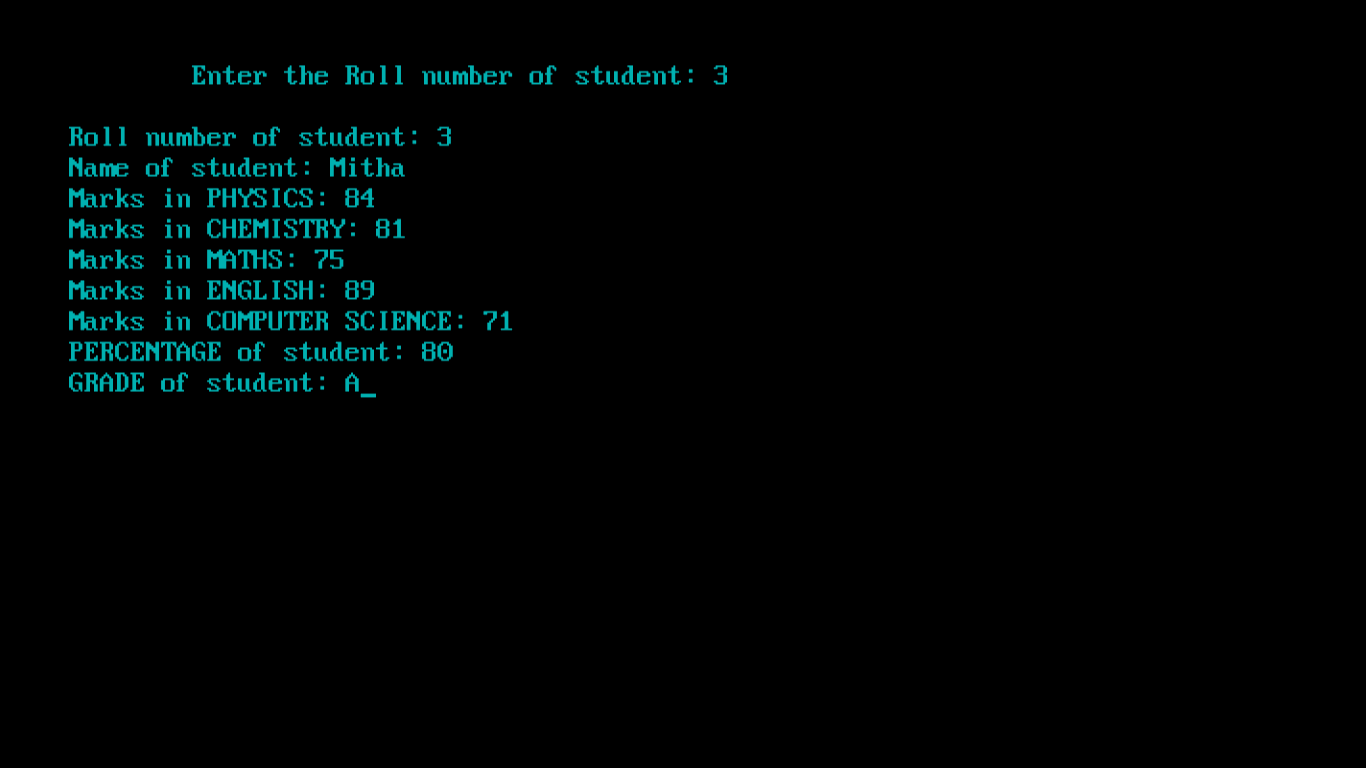






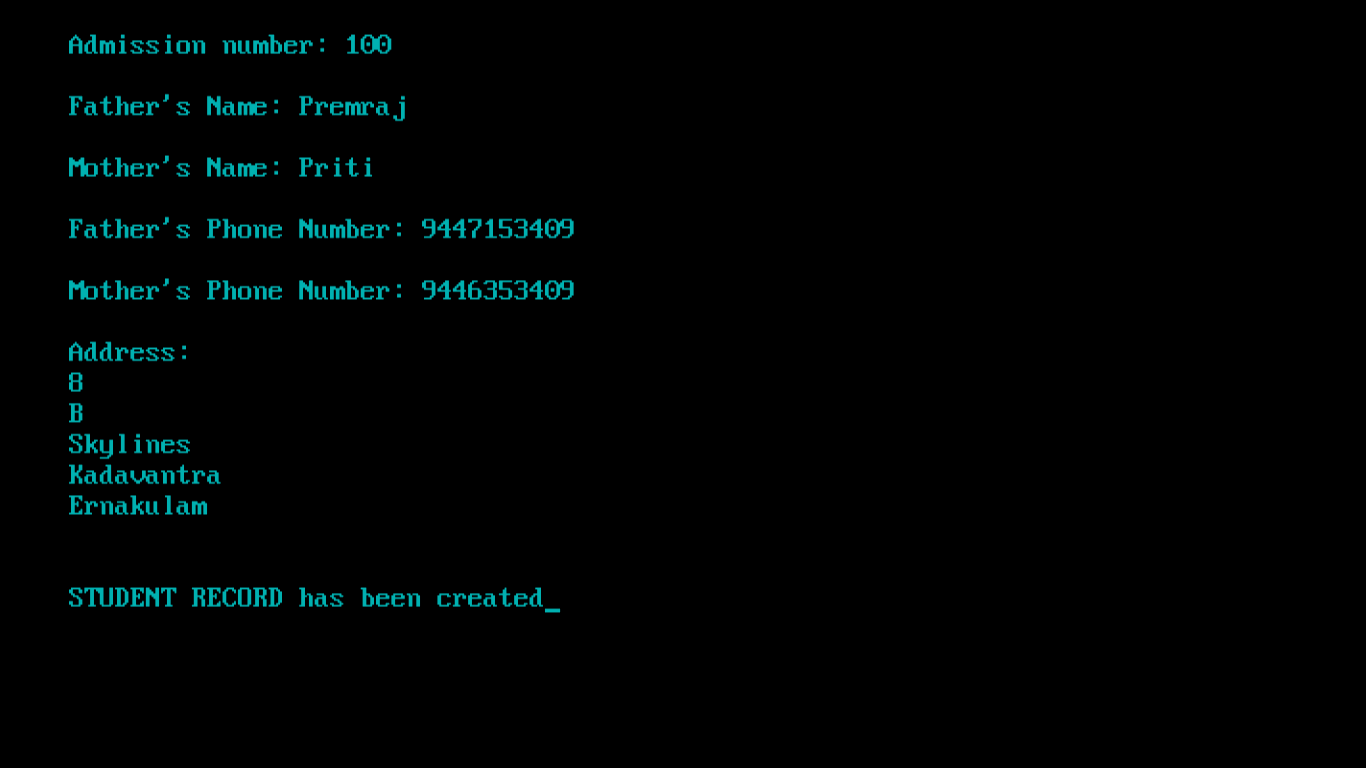




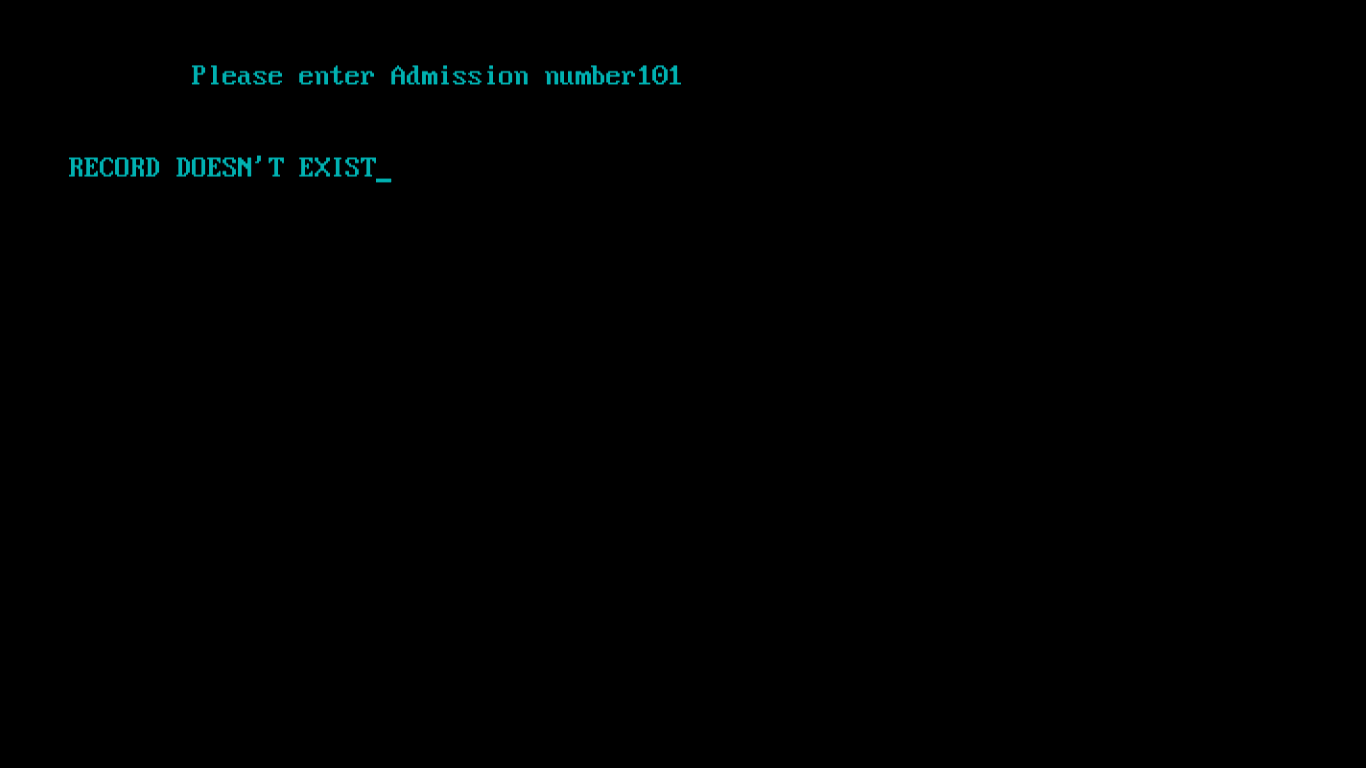




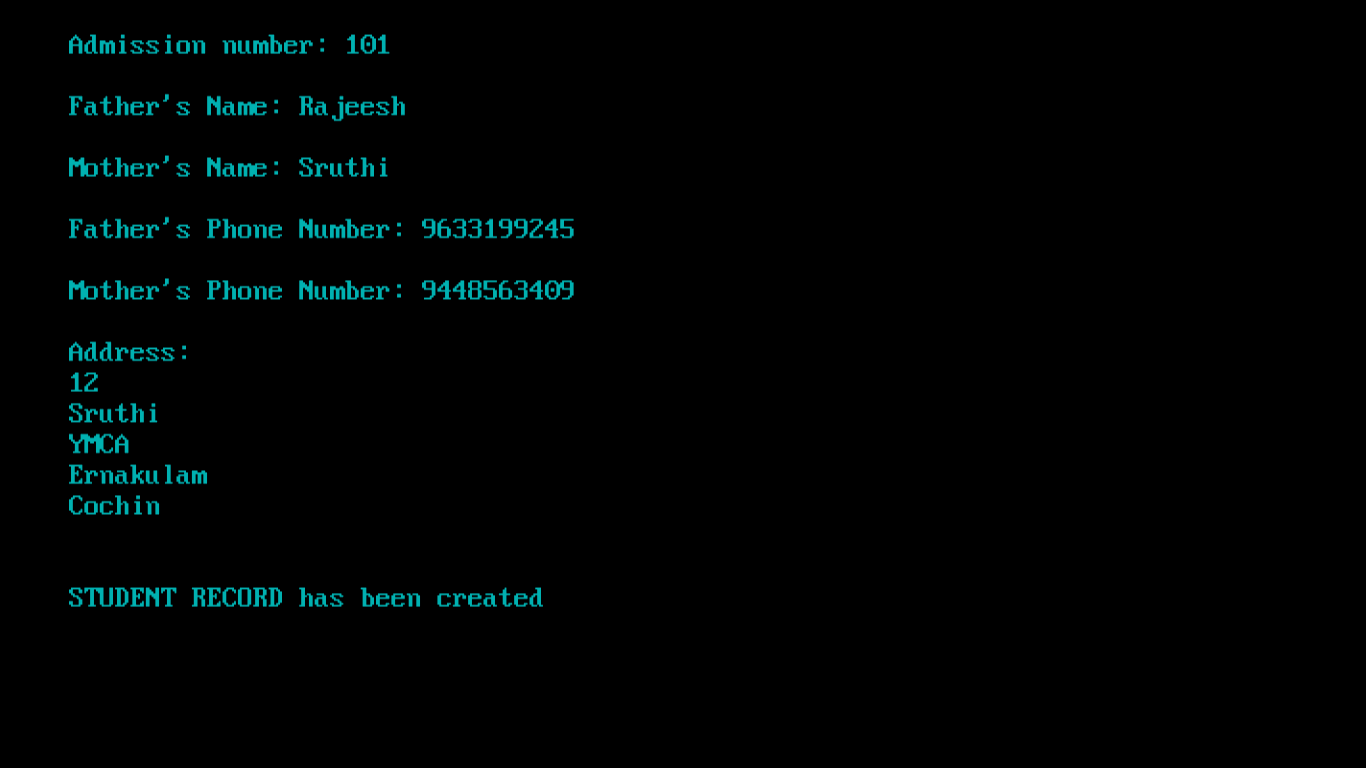




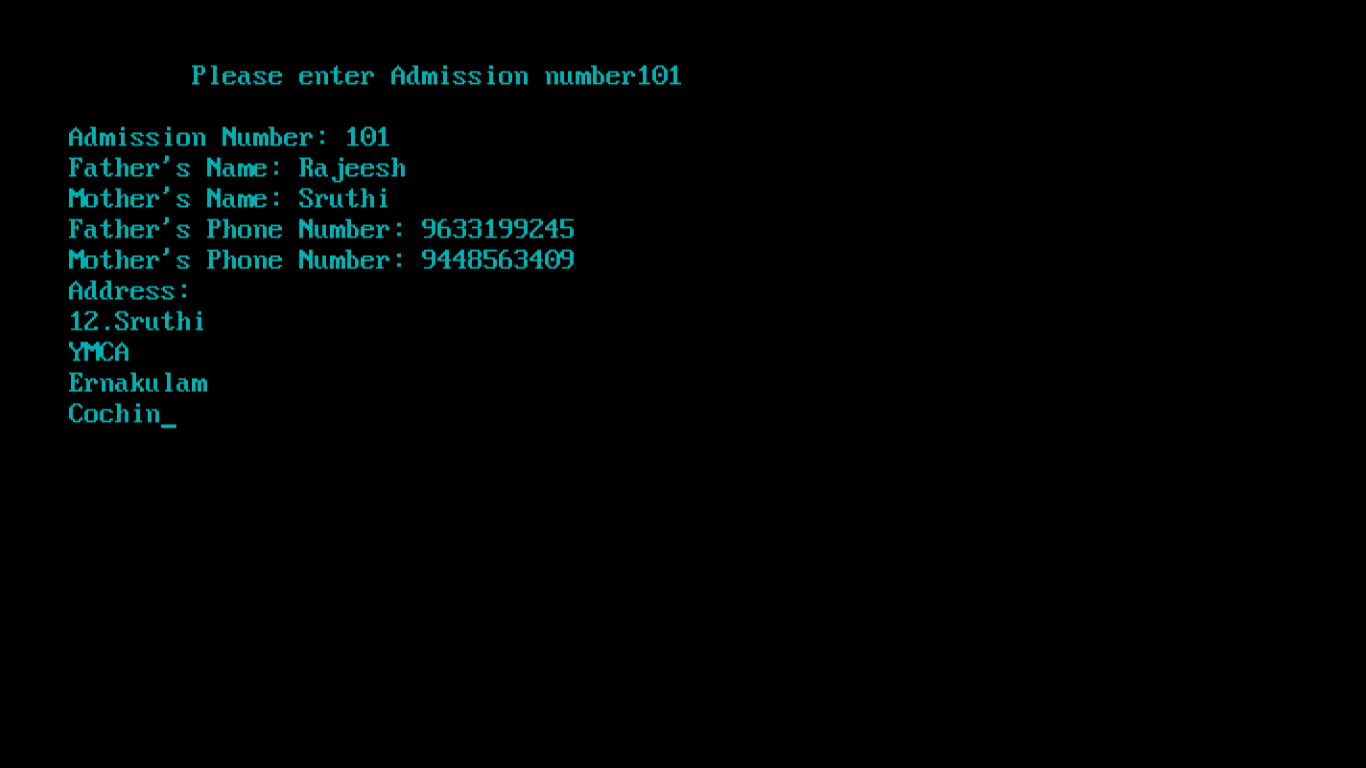


















BIBLIOGRAPHY

Computer Science with C++ for Class - 12

**Author:** Sumita Arora

**Publisher:** Dhanpat Rai Publications (P) Ltd (