



LAXMI INTERNATIONAL SCHOOL
P.B. No. 15, SARIGAM P.O., VALSAD – 396 155, GUJARAT, INDIA.

CERTIFICATE

This is to certify that the project work entitled “*LIBRARY MANAGEMENT SYSTEM*” is a bonafide work done by Miss YASHITA NAMDEO , Seat No: _____ ,

IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE PRACTICAL IN
COMPUTER SCIENCE(083) OF
CLASS XIIth SCIENCE

The project work was carried out under our supervision and is certified further that to the best of our knowledge, the work reported here does not form part of any other thesis or dissertation of any other candidate.

GUIDE

PRINCIPAL

Mr.Javed Shaikh

Laxmi International School
Sarigam

EXAMINER

ACKNOWLEDGEMENT

I whole heartily wish to thank Mr. Shaji Mathew sir, for his immense help in carrying out this project.

My special thanks to Mr.Javed Shaikh for his valuable guidance in our project work.

I would like to take this opportunity to thank the campus director Sir , Mr.Shridhar and all staff members of Senior Secondary Wing, LAXMI INTERNATIONAL SCHOOL, SARIGAM for their effort in building my career.

Without the support and encouragement of our parents, relatives and friends this project work would never have become a reality.

Yashita Namdeo



INDEX

<i>Sr. No.</i>	<i>TITLE</i>	<i>Page No.</i>	<i>Sign.</i>
<i>1.</i>	<i>Introduction to C++</i>	<i>6</i>	
<i>2.</i>	<i>Introduction to Project</i>	<i>8</i>	
<i>3.</i>	<i>System Requirements</i>	<i>10</i>	
<i>4.</i>	<i>Headers Files & Functions</i>	<i>12</i>	
<i>6.</i>	<i>Source Code</i>	<i>14</i>	
<i>7.</i>	<i>Output screens</i>	<i>28</i>	
<i>8.</i>	<i>Bibliography</i>	<i>39</i>	



INTRODUCTION TO C++

- C++, as we all know is an extension to C language and was developed by Bjarne Stroustrup of AT&T Bell Laboratories in the early 1980's.
- The “++” is a syntactic constructed in C (to increment a variable), and C++, so that most C programs can be compiled using a C++ compiler.
- C++ is an intermediate level language, as it comprises a confirmation of both high level and low level language features.
- C++ is a statically typed, free form, multiparadigm, compiled general-purpose language.
- C++ is an Object Oriented Programming language but is not purely Object Oriented.
- Its features like Friend and Virtual, violate some of the very important OOPS features, rendering this language unworthy of being called completely Object Oriented. It is a middle level language.

C++



Paradigm	Multi-paradigm: procedural, functional, object-oriented, generic ^[1]
Designed by	Bjarne Stroustrup
First appeared	1983; 34 years ago
Stable release	ISO/IEC 14882:2014 / 15 December 2014; 2 years ago
Typing discipline	Static, nominative, partially inferred
Implementation language	C++ or C
Filename extensions	.cc .cpp .cxx .C .c++ .h .hh .hpp .hxx .h++
Website	isocpp.org 

Major implementations
 LLVM Clang, GCC, Microsoft Visual C++, Embarcadero C++Builder, Intel C++ Compiler, IBM XL C++, EDG

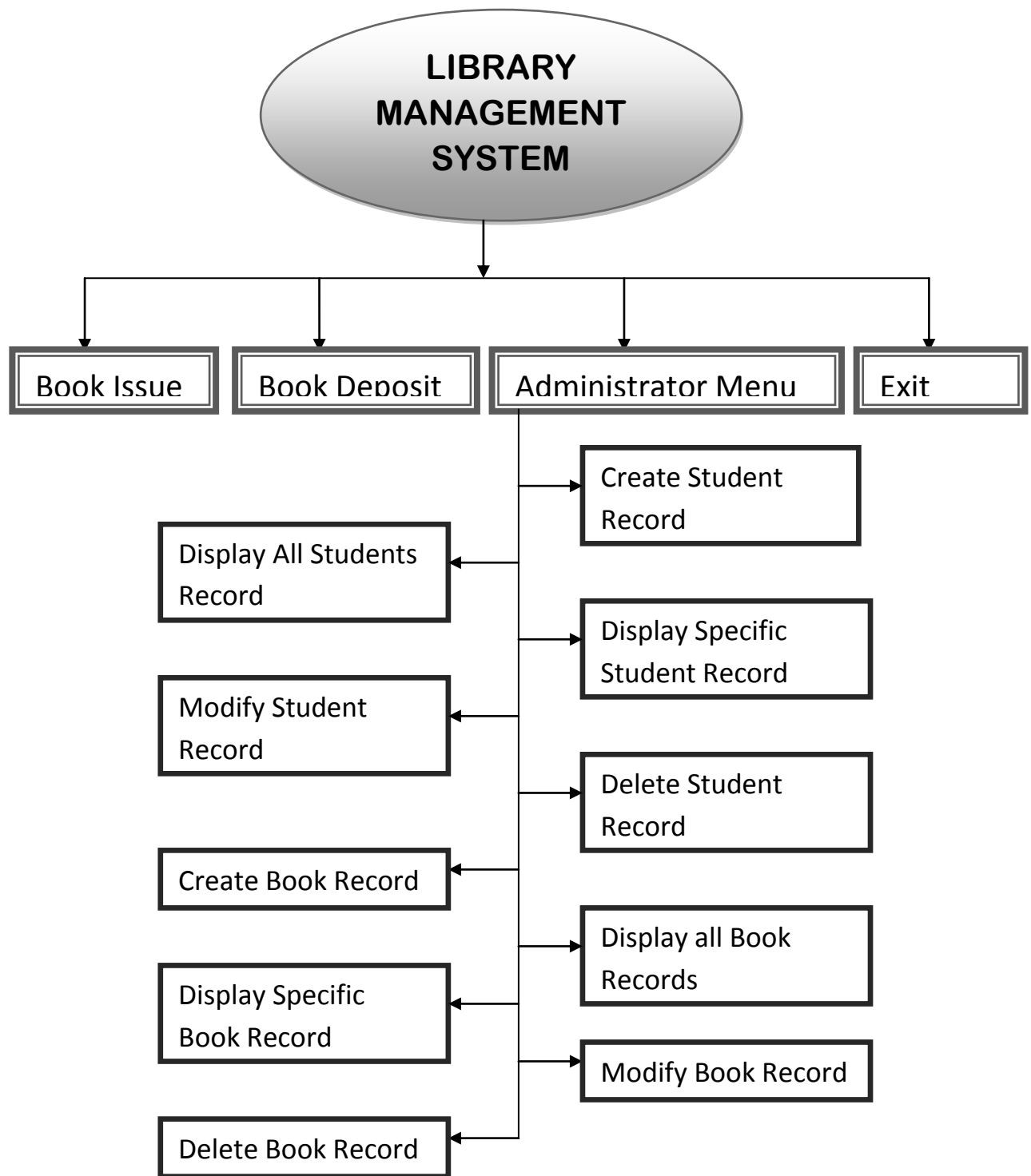
Influenced by
 Ada, ALGOL 68, C, CLU, ML, Simula, Python^{[2][3]}

Influenced
 Ada 95, C#,^[4] C99, Chapel,^[5] D, Java,^[6] Lua, Perl, PHP, Python,^[citation needed] Rust, Nim^[citation needed]

 C++ Programming at Wikibooks



INTRODUCTION TO PROJECT





SYSTEM REQUIREMENTS

View basic information about your computer

Windows edition

Windows 7 Ultimate

Copyright © 2009 Microsoft Corporation. All rights reserved.

Service Pack 1



System

Rating:



Windows Experience Index

Processor: Intel(R) Core(TM) i3 CPU M 350 @ 2.27GHz 2.27 GHz

Installed memory (RAM): 2.00 GB (1.86 GB usable)

System type: 32-bit Operating System

Pen and Touch: No Pen or Touch Input is available for this Display


Computer name, domain, and workgroup settings

Computer name: hp-PC

Full computer name: hp-PC

Computer description:

Workgroup: WORKGROUP

 [Change settings](#)

Windows activation

Status Not Available

Product ID: Not Available



HEADER FILES AND FUNCTIONS

- *fstream.h* : It is used for defining several iostreams template classes that manipulate external files *fstream* is a standard C++ library that handles reading from and writing to files either in text or in binary formats.
- *conio.h* : This header declares several useful library functions for performing "console input and output" from a program. The library functions declared by *conio.h* vary somewhat from compiler to compiler.
- *stdio.h* : This header file defines types and macros needed for the standard I/O package. This file also defines the standard I/O predefined streams i.e., *gets*, *puts*, etc..
- *process.h* : It is the header file which contains function declarations and macros used in working with threads and processes.
- *string.h* : This header file declares several string manipulation and memory manipulation routines.
- *iomanip.h* : This header file declares the C++ streams I/O manipulators and contains macros for creating parameterised manipulators.
- *ctype.h* : This header file declares several functions that are useful for testing and mapping character.



SOURCE

CODE

// SOURCE CODE STARTS...

// HEADER FILE

```
#include<fstream.h>
#include<conio.h>
#include<stdio.h>
#include<process.h>
#include<string.h>
#include<iomanip.h>
#include<ctype.h>
```

// CLASS USED

```
class book
{
    char bno[6];
    char bname[50];
    char aname[20];
public:
    void create_book()
    {
        cout<<"\nNEW BOOK ENTRY...\n";
        cout<<"\nEnter The book no.";
        cin>>bno;
        cout<<"\n\nEnter The Name of The Book ";
        gets(bname);
        cout<<"\n\nEnter The Author's Name ";
        gets(aname);
        cout<<"\n\nBook Created..";
    }
    void show_book()
    {
        cout<<"\nBook no. : "<<bno;
        cout<<"\nBook Name : ";
        puts(bname);
        cout<<"Author Name : ";
        puts(aname);
    }
    void modify_book()
    {
        cout<<"\nBook no. : "<<bno;
        cout<<"\nModify Book Name : ";
        gets(bname);
        cout<<"\nModify Author's Name of Book : ";
        gets(aname);
    }
}
```

```

char* retbno()
{
    return bno;
}
void report()
{
    cout<<bno<<setw(30)<<bname<<setw(30)<<aname<<endl;
}

};    //class book ends here

```

```

class student
{
    char admno[6];
    char name[20];
    char stbno[6];
    int token;
public:
    void create_student()
    {
        clrscr();
        cout<<"\nNEW STUDENT ENTRY...\n";
        cout<<"\nEnter The admission no. ";
        cin>>admno;
        cout<<"\nEnter The Name of The Student ";
        gets(name);
        token=0;
        stbno[0]='\0';
        cout<<"\n\nStudent Record Created..";
    }
    void show_student()
    {
        cout<<"\nAdmission no. : "<<admno;
        cout<<"\nStudent Name : ";
        puts(name);
        cout<<"\nNo of Book issued : "<<token;
        if(token==1)
            cout<<"\nBook No "<<stbno;
    }
    void modify_student()
    {
        cout<<"\nAdmission no. : "<<admno;
        cout<<"\nModify Student Name : ";
        gets(name);
    }
    char* retadmno()
    {

```

```

        return admno;
    }
    char* retstbno()
    {
        return stbno;
    }
    int rettoken()
    {
        return token;
    }
    void addtoken()
    {
        token=1;
    }
    void resettoken()
    {
        token=0;
    }
    void getstbno(char t[])
    {
        strcpy(stbno,t);
    }
    void report()
    {
        cout<<"\t"<<admno<<setw(20)<<name<<setw(10)<<token<<endl;
    }

};    //class student ends here

```

// Global declaration for stream object, object

```

fstream fp,fp1;
book bk;
student st;

```

//Function to write in file

```

void write_book()
{
    char ch;
    fp.open("book.dat",ios::out|ios::app);
    do
    {
        clrscr();
        bk.create_book();
        fp.write((char*)&bk,sizeof(book));
        cout<<"\n\nDo you want to add more record..(y/n?)";
    }
}

```



```

        cin>>ch;
    }while(ch=='y' | ch=='Y');
    fp.close();
}
void write_student()
{
    char ch;
    fp.open("student.dat",ios::out|ios::app);

    do
    {
        st.create_student();
        fp.write((char*)&st,sizeof(student));
        cout<<"\n\ndo you want to add more record..(y/n?)";
        cin>>ch;
    }while(ch=='y' | ch=='Y');
    fp.close();
}

```

// function to read specific record from file

```

void display_spb(char n[])
{
    cout<<"\nBOOK DETAILS\n";
    int flag=0;
    fp.open("book.dat",ios::in);
    while(fp.read((char*)&bk,sizeof(book)))
    {
        if(strcmpi(bk.retbn(),n)==0)
        {
            bk.show_book();
            flag=1;
        }
    }
    fp.close();
    if(flag==0)
        cout<<"\n\nBook does not exist";
    getch();
}
void display_sps(char n[])
{
    cout<<"\nSTUDENT DETAILS\n";
    int flag=0;
    fp.open("student.dat",ios::in);
    while(fp.read((char*)&st,sizeof(student)))
    {
        if((strcmpi(st.retadmno(),n)==0))

```

```

    {
        st.show_student();
        flag=1;
    }
}
fp.close();
if(flag==0)
    cout<<"\n\nStudent does not exist";
getch();
}

```

// function to modify record of file

```

void modify_book()
{
    char n[6];
    int found=0;
    clrscr();
    cout<<"\n\n\tMODIFY BOOK RECORD.... ";
    cout<<"\n\n\tEnter The book no. of The book";
    cin>>n;
    fp.open("book.dat",ios::in|ios::out);
    while(fp.read((char*)&bk,sizeof(book)) && found==0)
    {
        if(strcmpi(bk.retbno(),n)==0)
        {
            bk.show_book();
            cout<<"\nEnter The New Details of book"<<endl;
            bk.modify_book();
            int pos=-1*sizeof(bk);
            fp.seekp(pos,ios::cur);
            fp.write((char*)&bk,sizeof(book));
            cout<<"\n\n\tRecord Updated";
            found=1;
        }
    }
    fp.close();
    if(found==0)
        cout<<"\n\nRecord Not Found ";
    getch();
}

```

```

void modify_student()
{
    char n[6];
    int found=0;
    clrscr();

```

```

cout<<"\n\n\tMODIFY STUDENT RECORD... ";
cout<<"\n\n\tEnter The admission no. of The student";
cin>>n;
fp.open("student.dat",ios::in|ios::out);
while(fp.read((char*)&st,sizeof(student)) && found==0)
{
    if(strcmpi(st.retadmno(),n)==0)
    {
        st.show_student();
        cout<<"\nEnter The New Details of student"<<endl;
        st.modify_student();
        int pos=-1*sizeof(st);
        fp.seekp(pos,ios::cur);
        fp.write((char*)&st,sizeof(student));
        cout<<"\n\n\tRecord Updated";
        found=1;
    }
}
fp.close();
if(found==0)
cout<<"\n\nRecord Not Found ";
getch();
}

```

// function to delete record of file

```

void delete_student()
{
    char n[6];
    int flag=0;
    clrscr();
    cout<<"\n\n\n\tDELETE STUDENT... ";
    cout<<"\n\nEnter The admission no. of the Student You Want To Delete : ";
    cin>>n;
    fp.open("student.dat",ios::in|ios::out);
    fstream fp2;
    fp2.open("Temp.dat",ios::out);
    fp.seekg(0,ios::beg);
    while(fp.read((char*)&st,sizeof(student)))
    {
        if(strcmpi(st.retadmno(),n)!=0)
            fp2.write((char*)&st,sizeof(student));
        else
            flag=1;
    }
    fp2.close();
    fp.close();
}

```

```

remove("student.dat");
rename("Temp.dat","student.dat");
if(flag==1)
    cout<<"\n\n\tRecord Deleted ..";
else
    cout<<"\n\nRecord not found";
getch();
}

void delete_book()
{
    char n[6];
    clrscr();
    cout<<"\n\n\n\tDELETE BOOK ...";
    cout<<"\n\nEnter The Book no. of the Book You Want To Delete : ";
    cin>>n;
    fp.open("book.dat",ios::in|ios::out);
    fstream fp2;
    fp2.open("Temp.dat",ios::out);
    fp.seekg(0,ios::beg);
    while(fp.read((char*)&bk,sizeof(book)))
    {
        if(strcmpi(bk.retbno(),n)!=0)
        {
            fp2.write((char*)&bk,sizeof(book));
        }
    }
    fp2.close();
    fp.close();
    remove("book.dat");
    rename("Temp.dat","book.dat");
    cout<<"\n\n\tRecord Deleted ..";
    getch();
}

```

//Function to display all students list

```

void display_alls()
{
    clrscr();
    fp.open("student.dat",ios::in);
    if(!fp)
    {
        cout<<"ERROR!!! FILE COULD NOT BE OPEN ";
        getch();
        return;
    }
}

```

```

cout<<"\n\n\t\tSTUDENT LIST\n\n";
cout<<"===== \n";
cout<<"\tAdmission No."<<setw(10)<<"Name"<<setw(20)<<"Book Issued\n";
cout<<"===== \n";
while(fp.read((char*)&st,sizeof(student)))
{
    st.report();
}
fp.close();
getch();
}

```

//Function to display Books list

```

void display_allb()
{
    clrscr();
    fp.open("book.dat",ios::in);
    if(!fp)
    {
        cout<<"ERROR!!! FILE COULD NOT BE OPEN ";
        getch();
        return;
    }
    cout<<"\n\n\t\tBook LIST\n\n";
    cout<<"===== \n";
    cout<<"Book Number"<<setw(20)<<"Book Name"<<setw(25)<<"Author\n";
    cout<<"===== \n";
    while(fp.read((char*)&bk,sizeof(book)))
    {
        bk.report();
    }
    fp.close();
    getch();
}

```

// function to issue book

```

void book_issue()
{
    char sn[6],bn[6];
    int found=0,flag=0;
    clrscr();
    cout<<"\n\nBOOK ISSUE ... ";
    cout<<"\n\n\tEnter The student's admission no. ";
    cin>>sn;
    fp.open("student.dat",ios::in|ios::out);
}

```

```

fp1.open("book.dat",ios::in|ios::out);
while(fp.read((char*)&st,sizeof(student)) && found==0)
{
    if(strcmpi(st.retadmno(),sn)==0)
    {
        found=1;
        if(st.rettoken()==0)
        {
            cout<<"\n\n\tEnter the book no. ";
            cin>>bn;
            while(fp1.read((char*)&bk,sizeof(book))&& flag==0)
            {
                if(strcmpi(bk.retbnno(),bn)==0)
                {
                    bk.show_book();
                    flag=1;
                    st.addtoken();
                    st.getstbnno(bk.retbnno());
                    int pos=-1*sizeof(st);
                    fp.seekp(pos,ios::cur);
                    fp.write((char*)&st,sizeof(student));
                    cout<<"\n\n\t Book issued successfully\n";
                    cout<<"\nPlease Note: Write the current date in backside of your book";
                    cout<<"and submit within 15 days fine Rs. 1 for each day after ";
                    cout<<"15 days period";
                }
            }
            if(flag==0)
                cout<<"Book no does not exist";
        }
        else
            cout<<"You have not returned the last book ";
    }
}
if(found==0)
    cout<<"Student record not exist...";
getch();
fp.close();
fp1.close();
}

```

// function to deposit book

```

void book_deposit()
{
    char sn[6],bn[6];
    int found=0,flag=0,day,fine;

```

```

clrscr();
cout<<"\n\nBOOK DEPOSIT ...";
cout<<"\n\n\tEnter The student's admission no.";
cin>>sn;
fp.open("student.dat",ios::in|ios::out);
fp1.open("book.dat",ios::in|ios::out);
while(fp.read((char*)&st,sizeof(student)) && found==0)
{
    if(strcmpi(st.retadmno(),sn)==0)
    {
        found=1;
        if(st.rettoken()==1)
        {
            while(fp1.read((char*)&bk,sizeof(book))&& flag==0)
            {
                if(strcmpi(bk.retbno(),st.retstbno())==0)
                {
                    bk.show_book();
                    flag=1;
                    cout<<"\n\nBook deposited in no. of days";
                    cin>>day;
                    if(day>15)
                    {
                        fine=(day-15)*1;
                        cout<<"\n\nFine has to deposited Rs. "<<fine;
                    }
                    st.resettoken();
                    int pos=-1*sizeof(st);
                    fp.seekp(pos,ios::cur);
                    fp.write((char*)&st,sizeof(student));
                    cout<<"\n\n\t Book deposited successfully";
                }
            }
            if(flag==0)
                cout<<"Book no does not exist";
        }
        else
            cout<<"No book is issued..please check!!";
    }
}
if(found==0)
    cout<<"Student record not exist...";
getch();
fp.close();
fp1.close();
}

```

// INTRODUCTION FUNCTION

```
void intro()
{
    clrscr();
    cout<<"\n ++++++";
    cout<<"\n +*****+";
    cout<<"\n +*****+";
    cout<<"\n +*****+";
    cout<<"\n +*****+";
    cout<<"\n +*****+";
    cout<<"\n +*****+";
    cout<<"\n +*****WELCOME*****+";
    cout<<"\n +*****+";
    cout<<"\n +*****+";
    cout<<"\n +*****+";
    cout<<"\n +*****+";
    cout<<"\n +*****+";
    cout<<"\n ++++++*+++++";
    cout<<"\n\n\t\t\t LIBRARY";
    cout<<"\n\n\t\t\t MANAGEMENT";
    cout<<"\n\n\t\t\t PROJECT";
    cout<<"\n MADE BY :YASHITA NAMDEO";
    cout<<"\n CLASS 12-A";
    cout<<"\n SHOOOL : LAXMI INTERNATIONAL SCHOOL";
    cout<<"\n \t\t\t\t\t Press any key to continue";
    getch();
}
```

```
// ADMINISTRATOR MENU FUNCTION
```

```
void admin_menu()
{
    clrscr();
    int ch2;
    cout<<"\n\n\n\tADMINISTRATOR MENU";
    cout<<"\n\n\t1.CREATE STUDENT RECORD";
    cout<<"\n\n\t2.DISPLAY ALL STUDENTS RECORD";
    cout<<"\n\n\t3.DISPLAY SPECIFIC STUDENT RECORD ";
    cout<<"\n\n\t4.MODIFY STUDENT RECORD";
    cout<<"\n\n\t5.DELETE STUDENT RECORD";
    cout<<"\n\n\t6.CREATE BOOK ";
    cout<<"\n\n\t7.DISPLAY ALL BOOKS ";
    cout<<"\n\n\t8.DISPLAY SPECIFIC BOOK ";
    cout<<"\n\n\t9.MODIFY BOOK ";
    cout<<"\n\n\t10.DELETE BOOK ";
    cout<<"\n\n\t11.BACK TO MAIN MENU";
}
```



```

cout<<"\n\n\tPlease Enter Your Choice (1-11) ";
cin>>ch2;
switch(ch2)
{
    case 1: clrscr();
            write_student();
            break;
    case 2: display_all();
            break;
    case 3: char num[6];
            clrscr();
            cout<<"\n\n\tPlease Enter The Admission No. ";
            cin>>num;
            display_sps(num);
            break;
    case 4: modify_student();
            break;
    case 5: delete_student();
            break;
    case 6: clrscr();
            write_book();
            break;
    case 7: display_allb();
            break;
    case 8:
        {
            char num[6];
            clrscr();
            cout<<"\n\n\tPlease Enter The book No. ";
            cin>>num;
            display_spb(num);
            break;
        }
    case 9: modify_book();
            break;
    case 10: delete_book();
            break;
    case 11 :return;
    default:cout<<"\a";
}
admin_menu();
}

```

// THE MAIN FUNCTION OF PROGRAM

```

void main()
{

```

```

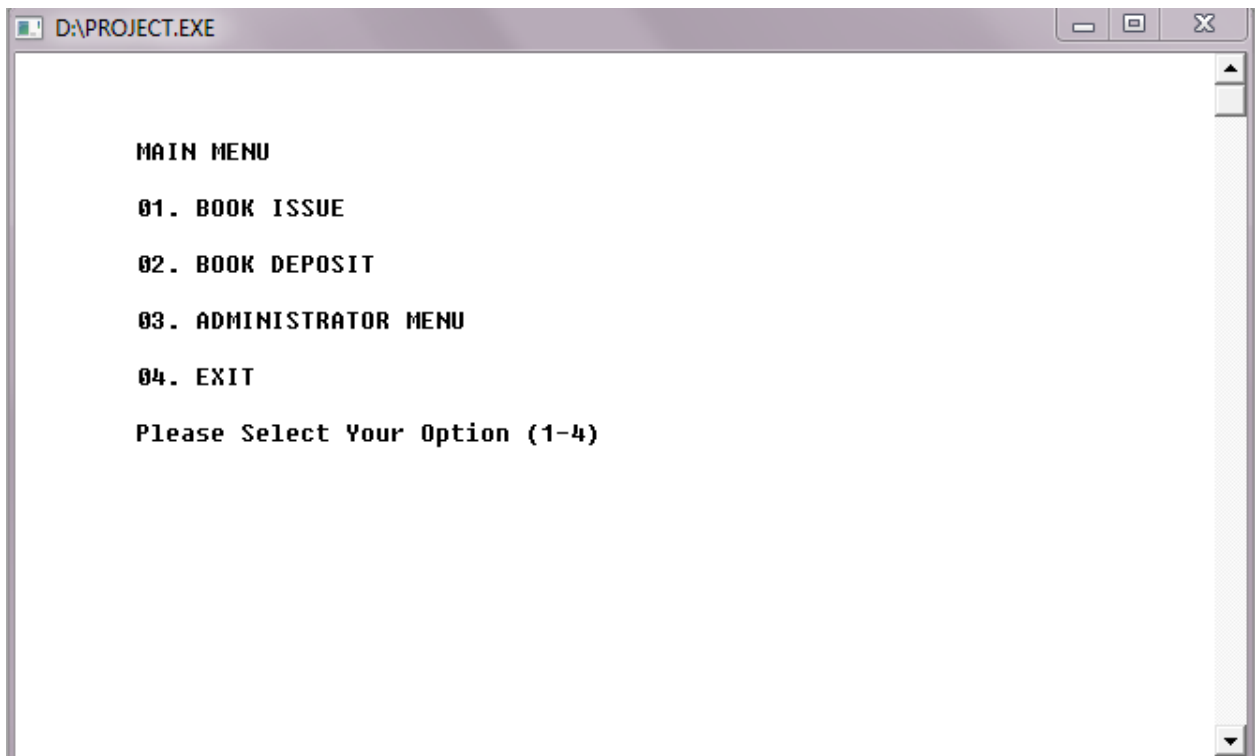
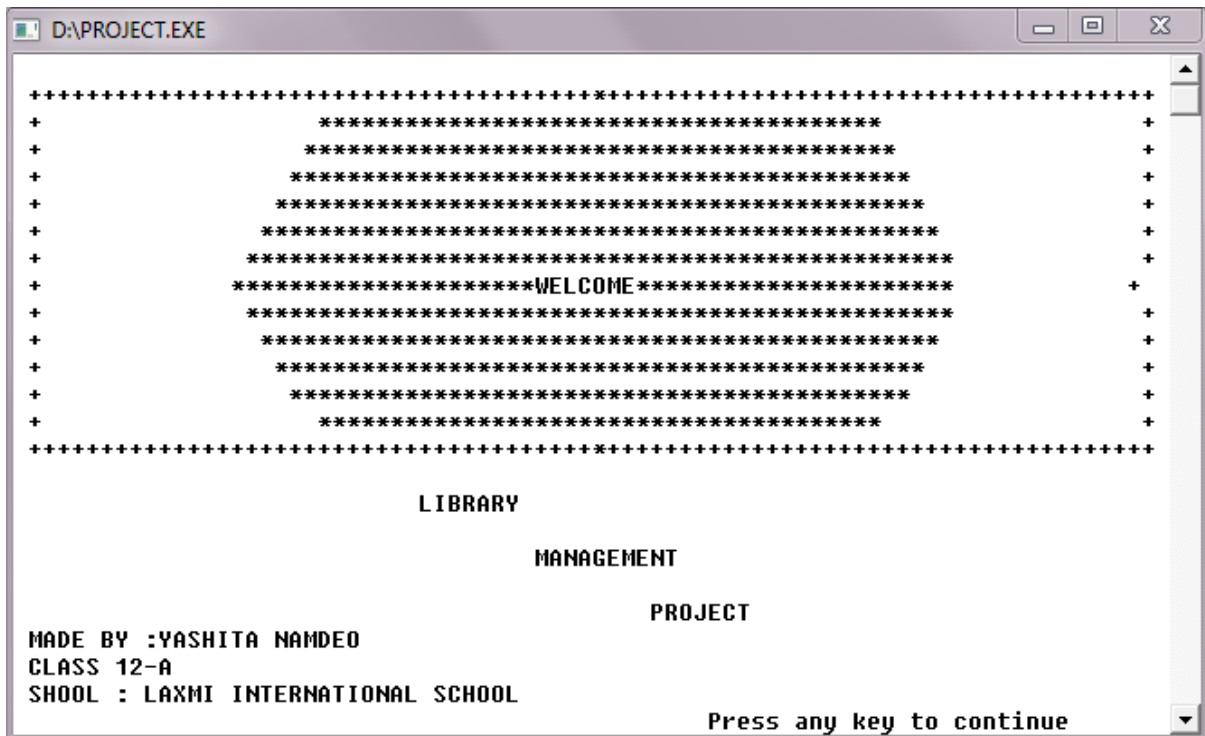
char ch;
intro();
do
{
    clrscr();
    cout<<"\n\n\n\tMAIN MENU";
    cout<<"\n\n\t01. BOOK ISSUE";
    cout<<"\n\n\t02. BOOK DEPOSIT";
    cout<<"\n\n\t03. ADMINISTRATOR MENU";
    cout<<"\n\n\t04. EXIT";
    cout<<"\n\n\tPlease Select Your Option (1-4) ";
    ch=getche();
    switch(ch)
    {
        case '1':clrscr();
        book_issue();
        break;
        case '2':book_deposit();
        break;
        case '3':admin_menu();
        break;
        case '4':exit(0);
        default :cout<<"\a";
    }
}while(ch!='4');
}

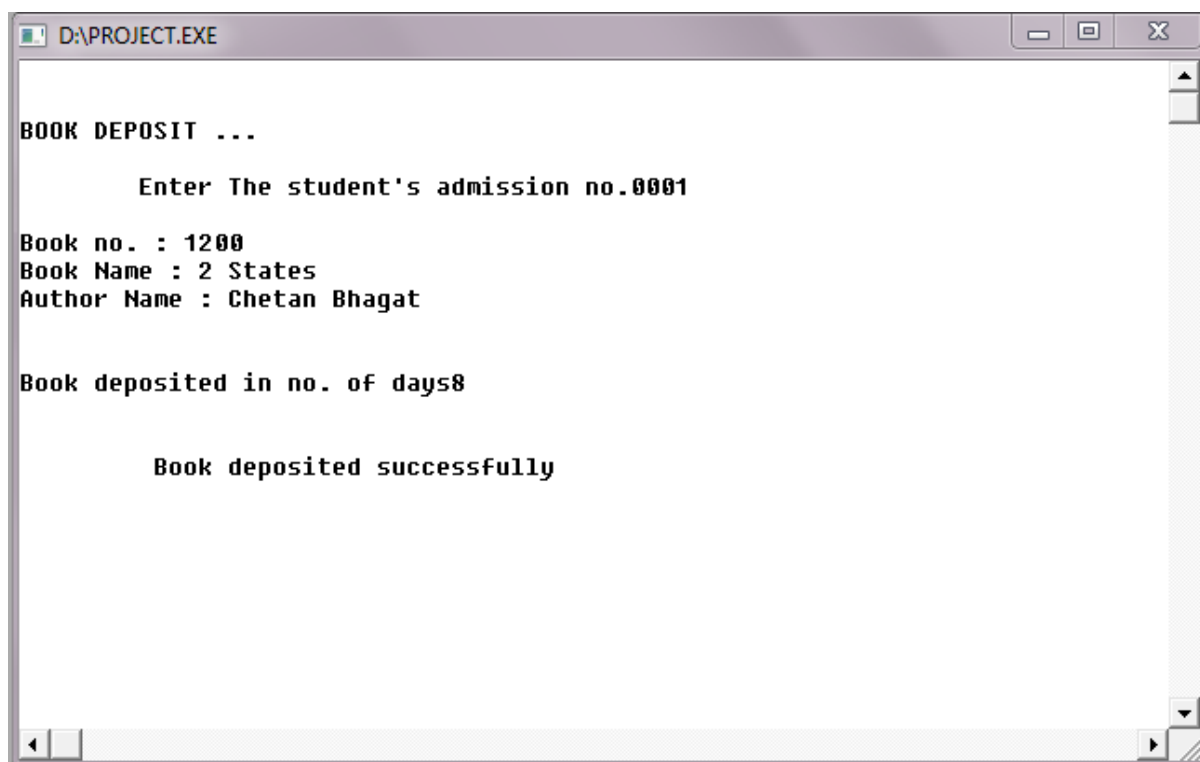
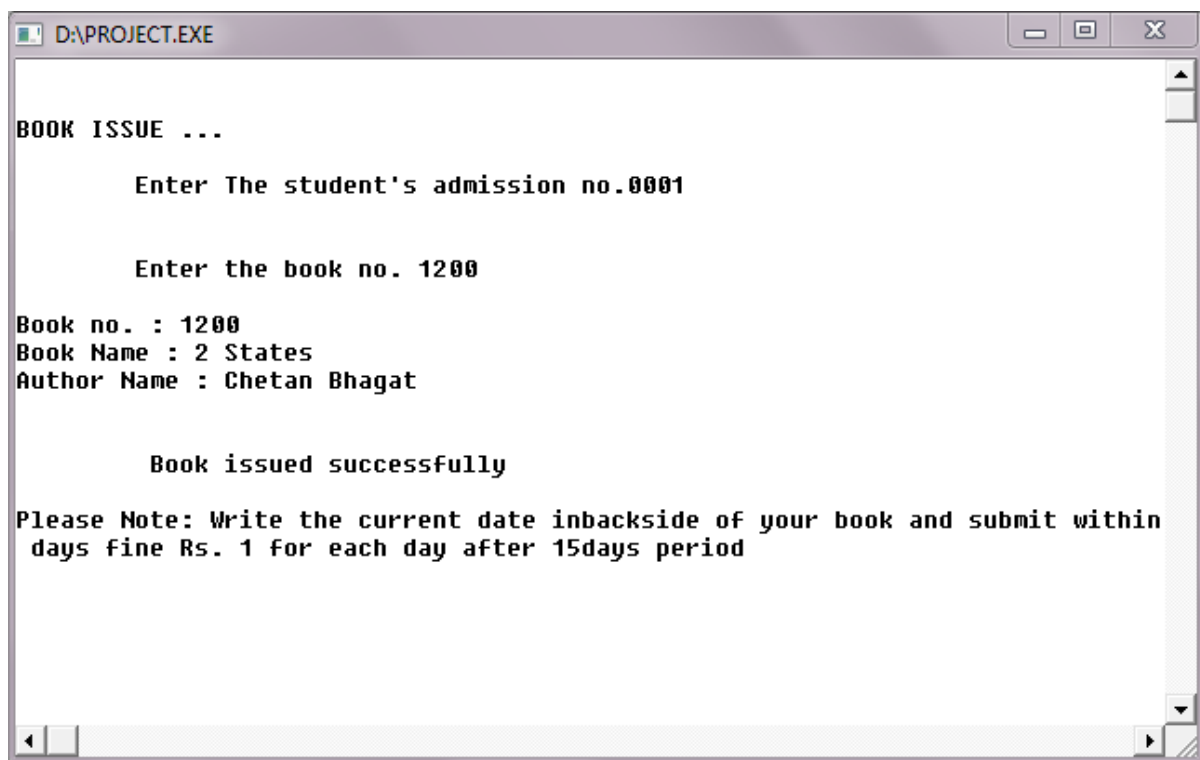
```

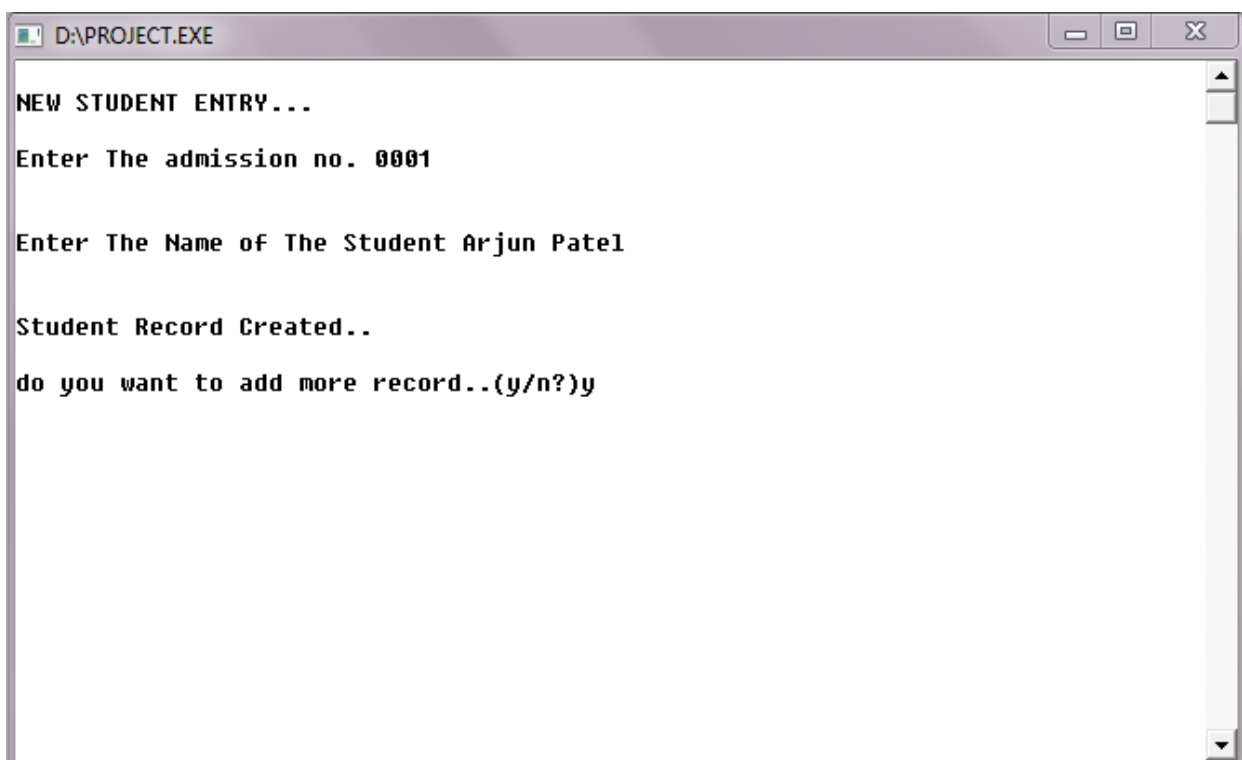
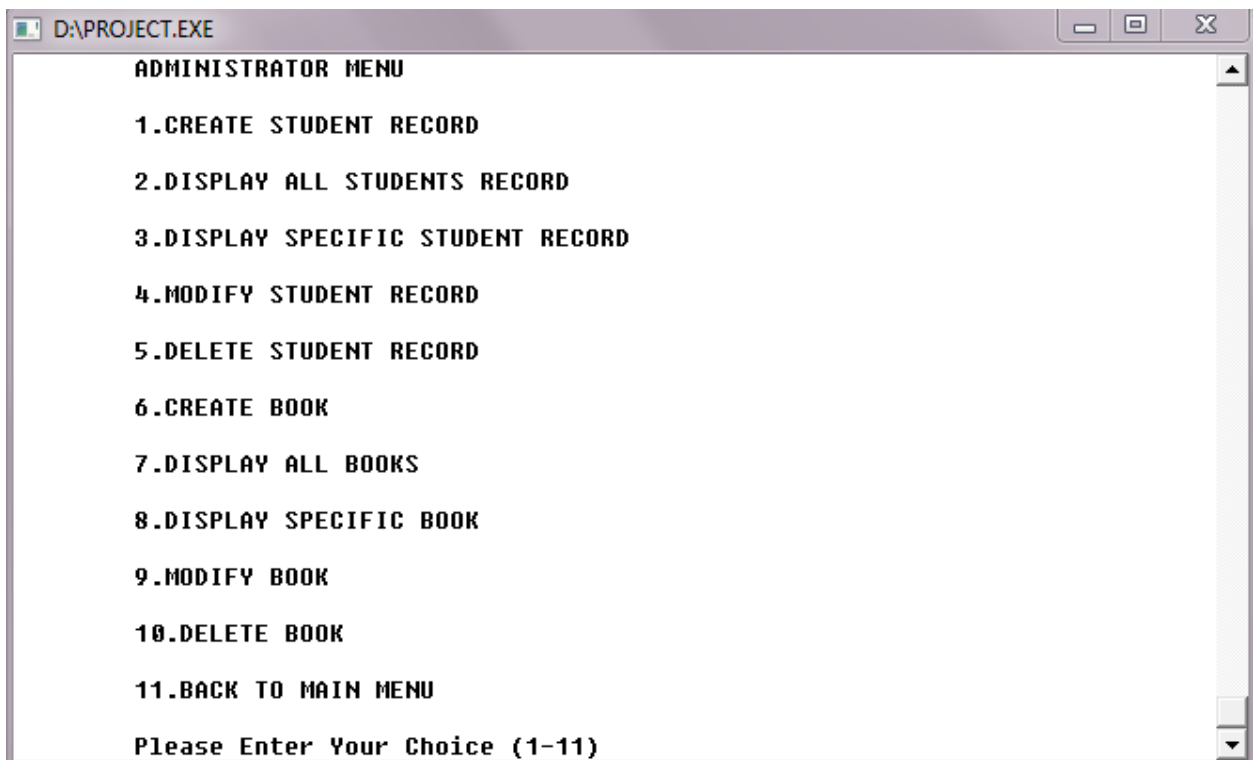
// SOURCE CODE ENDS...



OUTPUT SCREENS







D:\PROJECT.EXE

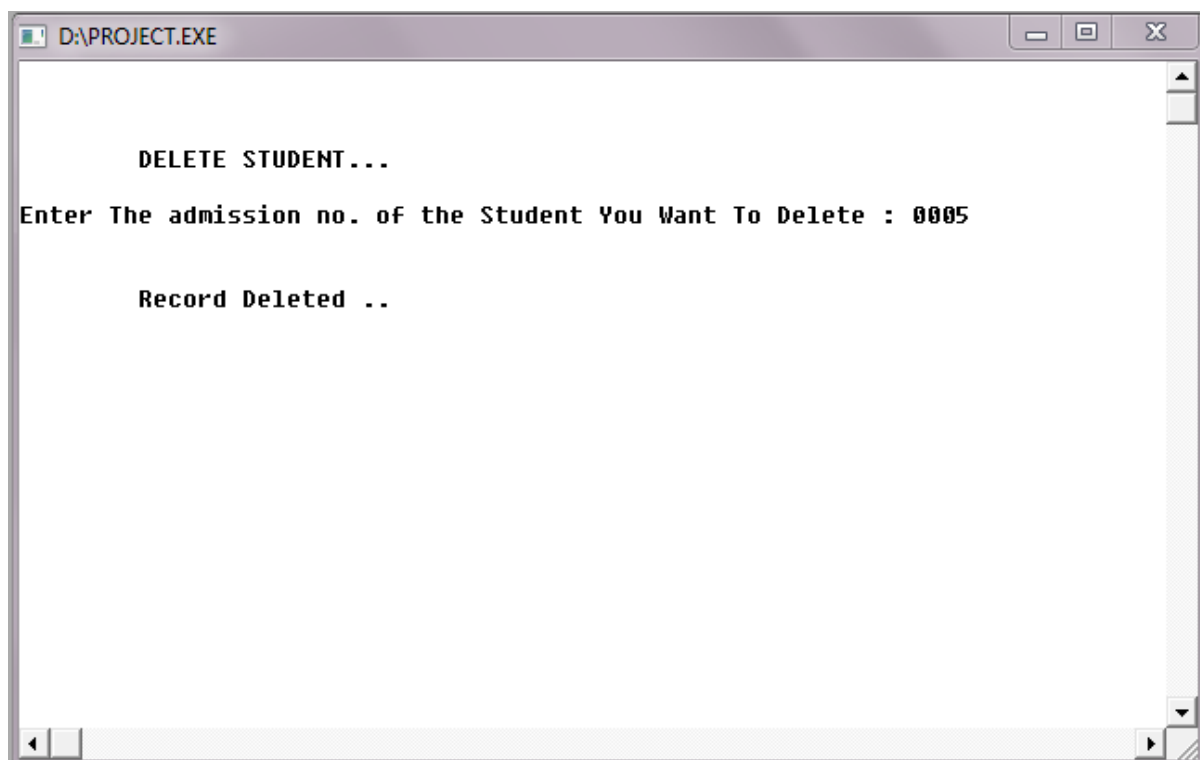
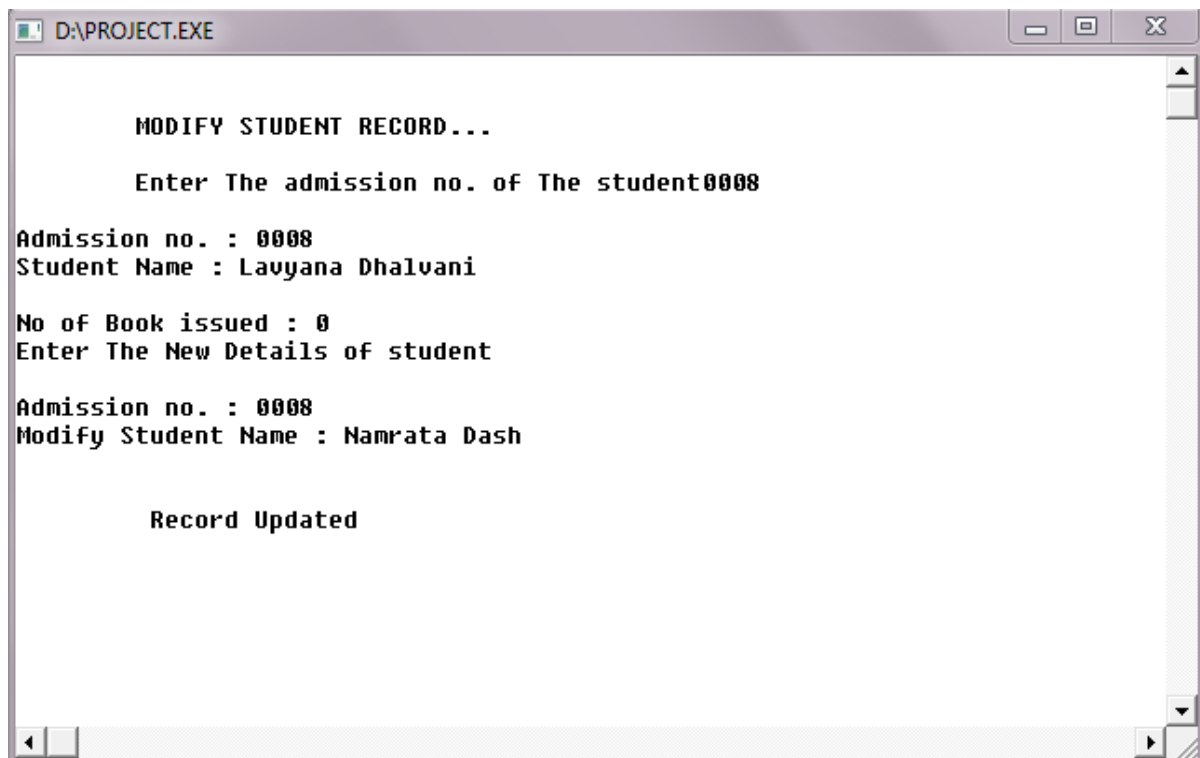
STUDENT LIST		
Admission No.	Name	Book Issued
0001	Arjun Patel	0
0002	Bhavik Sen	0
0003	Dhwani Shah	0
0004	Freya Shah	0
0005	Honey Dixit	0
0006	Kreena Patil	0
0008	Lavyana Dhalvani	0
0009	Mayank Sharma	0
0010	Taru Saini	0

D:\PROJECT.EXE

Please Enter The Admission No. 0001

STUDENT DETAILS

Admission no. : 0001
 Student Name : Arjun Patel
 No of Book issued : 0



D:\PROJECT.EXE

STUDENT LIST

Admission No.	Name	Book Issued
0001	Arjun Patel	0
0002	Bhavik Sen	0
0003	Dhwani Shah	0
0004	Freya Shah	0
0006	Kreena Patil	0
0008	Namrata Dash	0
0009	Mayank Sharma	0
0010	Taru Saini	0

D:\PROJECT.EXE

NEW BOOK ENTRY...

Enter The book no.6700

Enter The Name of The Book Pink Pages

Enter The Author's Name Sarah Delmege

Book Created..

Do you want to add more record..(y/n?)y

D:\PROJECT.EXE

Book LIST

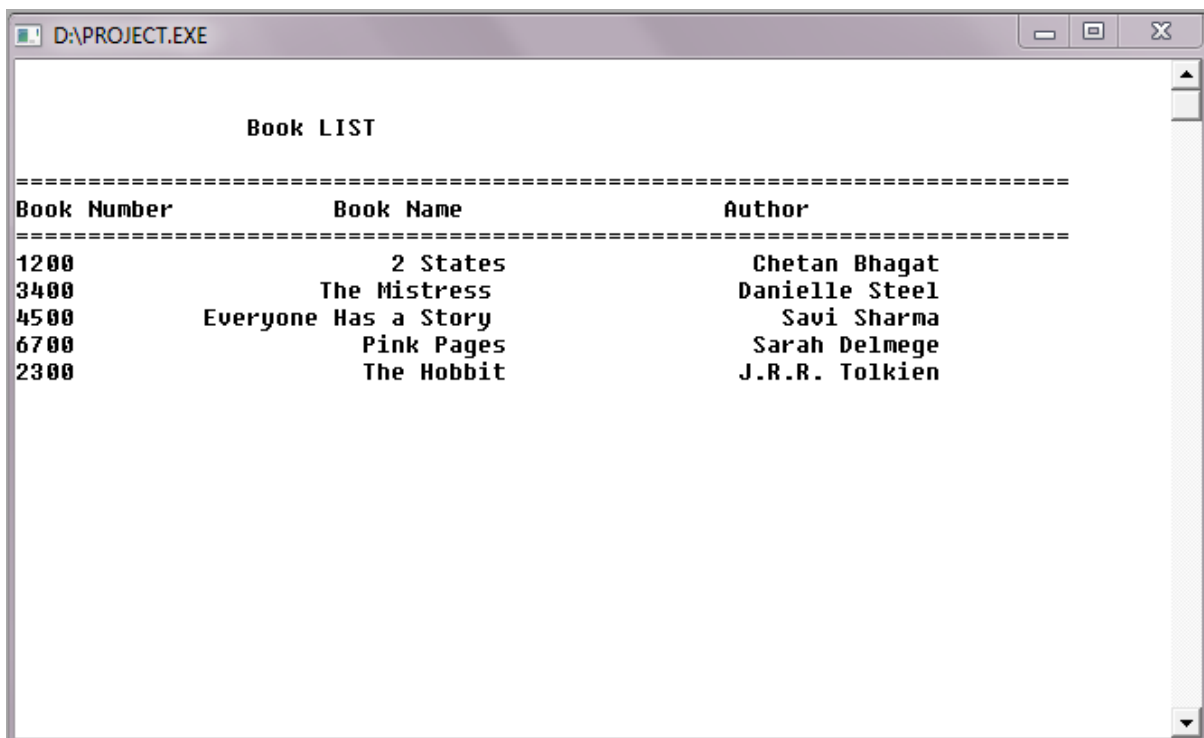
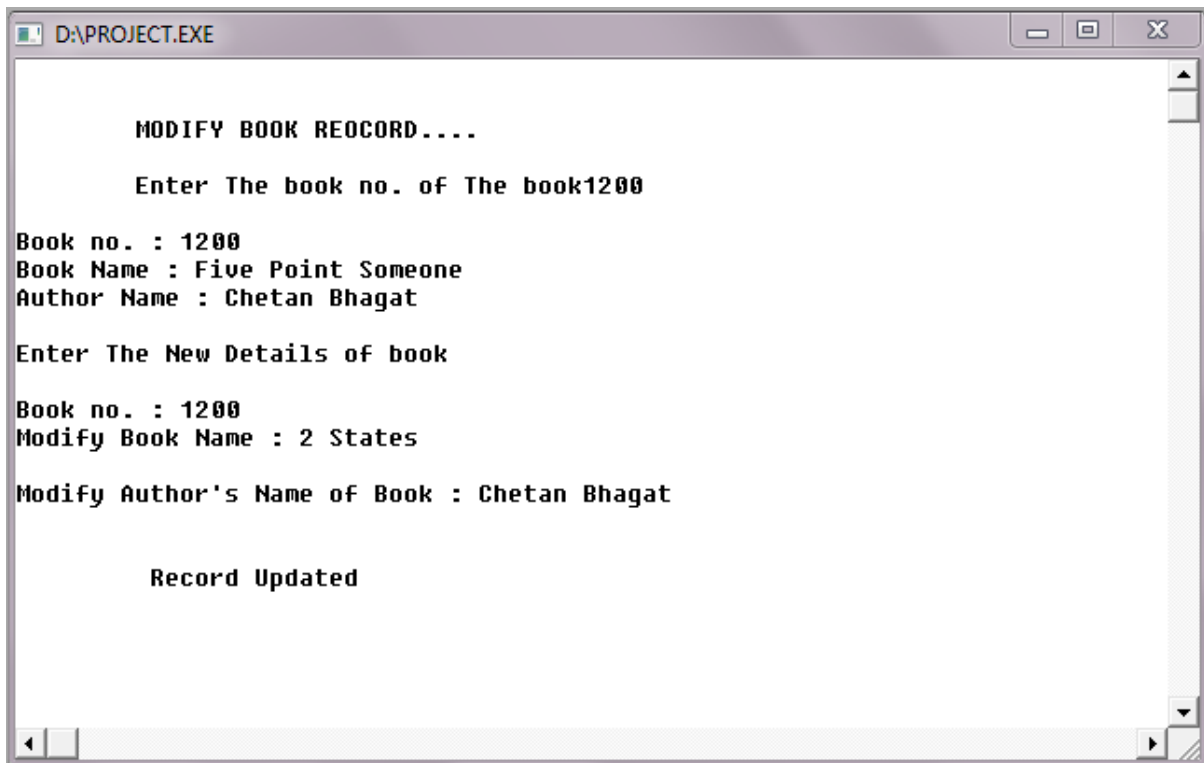
Book Number	Book Name	Author
1200	Five Point Someone	Chetan Bhagat
3400	The Mistress	Danielle Steel
4500	Everyone Has a Story	Savi Sharma
6700	Pink Pages	Sarah Delmege
2300	The Hobbit	J.R.R Tolkien

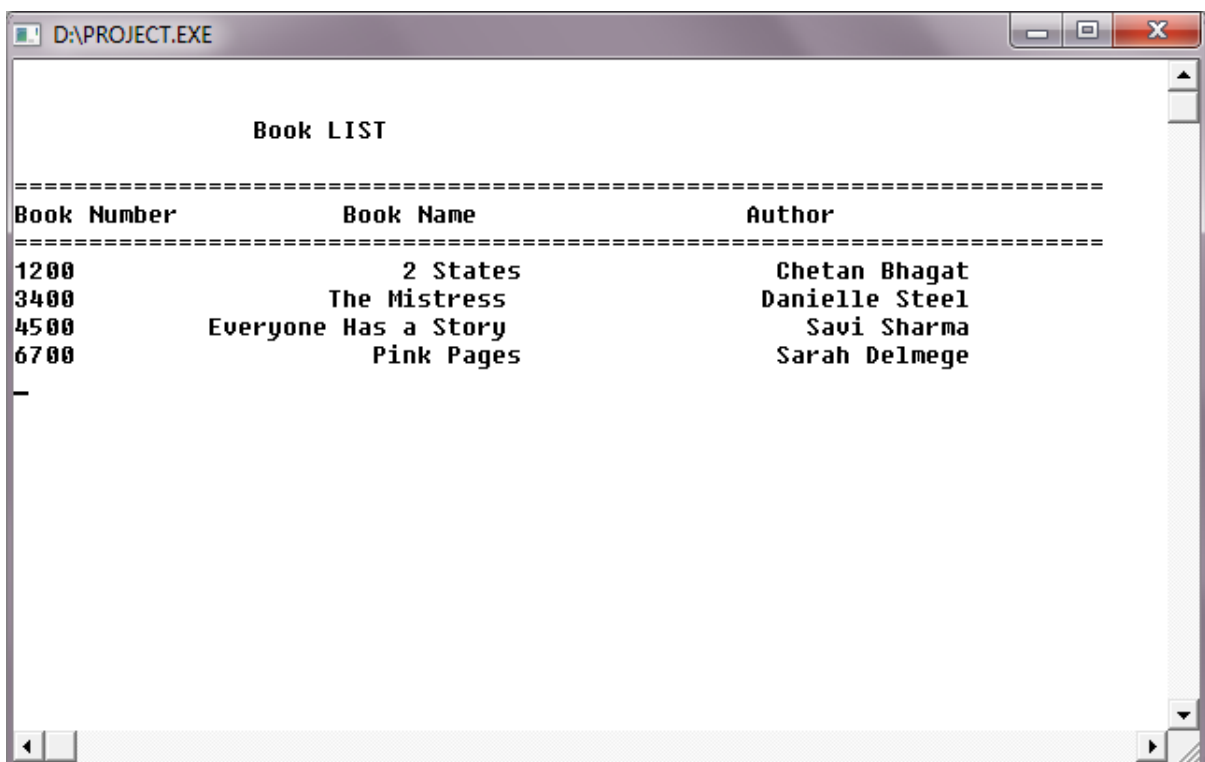
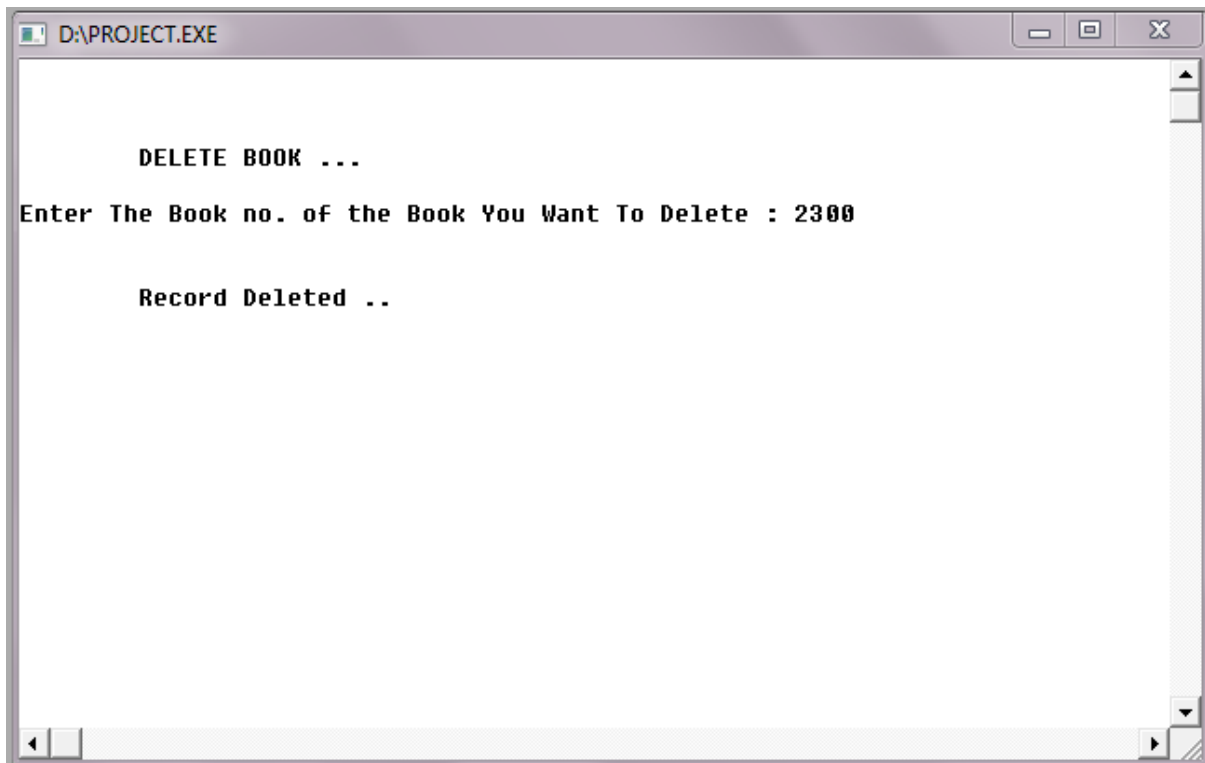
D:\PROJECT.EXE

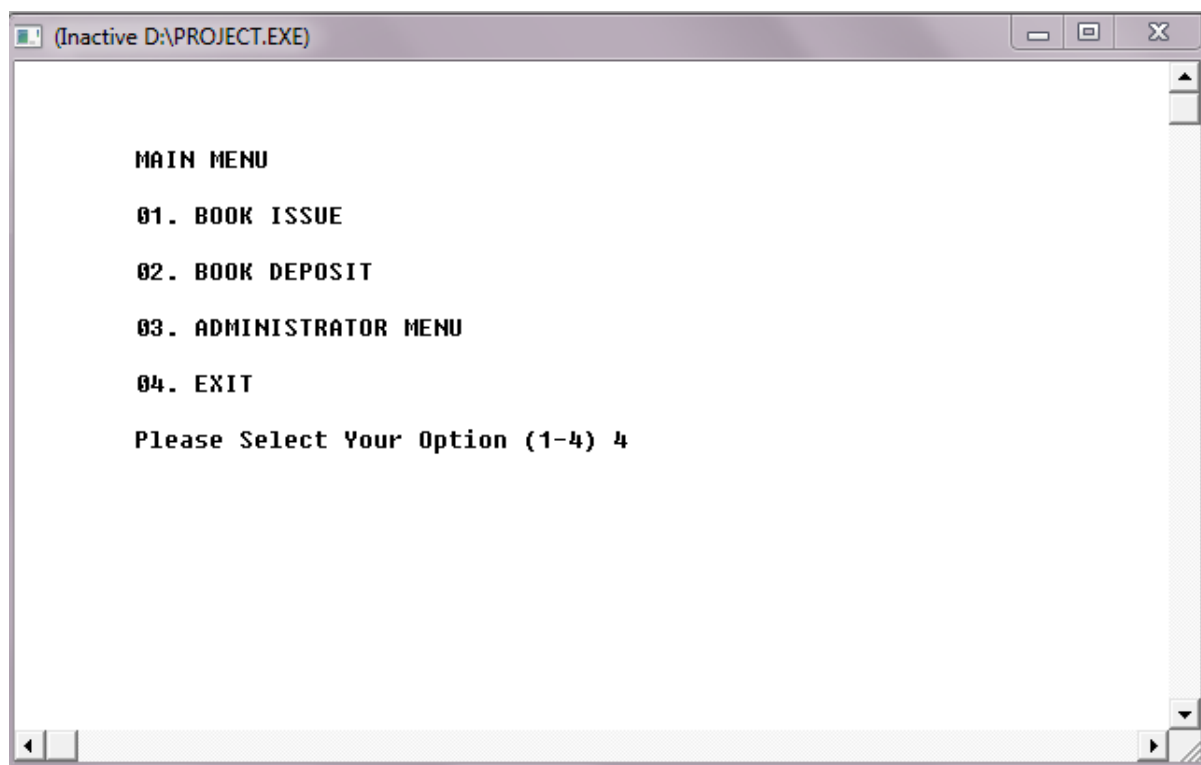
Please Enter The book No. 1200

BOOK DETAILS

Book no. : 1200
Book Name : Five Point Someone
Author Name : Chetan Bhagat









BIBLIOGRAPHY

- ✓ *Saraswati Computer Science with C++*
- ✓ *Computer Science with C++, by Sumita Arora*
- ✓ www.codeforc++indatafilehandling.com
- ✓ www.historyofc++.com
- ✓ www.headerfilesc++.com
- ✓ www.sourcecodes_librarymanagementsyateminc++.com
- ✓ www.easycodingsforprojectclass12c++.com