Project-I Report on

**LIBRARY MANAGEMENT SYSTEM**

*submitted towards the complete fulfillment of the*

*requirement for the award of the degree of*

**Bachelor of Technology**

**Software Engineering**

*Submitted by:*

**Student1**

**2K20/A8/02**

**and**

**Student2**

**2K20/A8/12**

Under the Supervision

*Of*

Dr.Prashant Giridhar Shambharkar



**Department of Computer Engineering** Delhi

Technological University Bawana Road.

Delhi -110042

July -2021

**DECLARATION**

We hereby certify that the work, which is presented in the Project-I entitled **Library Managemen**t in fulfillment of the requirement for the award of the Degree of Bachelor of Technology in Software Engineering and submitted to the Department of Computer Engineering, Delhi Technological University, Delhi is an authentic record of my/our own, carried out under the supervision of Dr.Prashant Giridhar Shambharkar.

The work presented in this report has not been submitted and not under consideration for the award for any other course/degree of this or any other Institute/University.

|  |  |
| --- | --- |
| **Student1** | **Student2** |
|  |  |
| 2K20/A8/02 | 2K20/A8/12 |
|  | B.Tech, Software |
| B.Tech, |
| Software | Engineering |
| Engineering |  |
|  |  |

**SUPERVISOR**

**CERTIFICATE**

To the best of my knowledge, the report comprises original work and has not been submitted in part or full for any Course/Degree to this university or elsewhere as per the candidate’s declaration.

Place: Delhi Dr.**Prashant Giridhar**

**ACKNOWLEDGEMENT**

In performing our major project, we had to take the help and guidance of some respected people, who deserve our greatest gratitude. The completion of this assignment gives us much pleasure. We would like to show our gratitude towards **DR. Prashant Giridhar**, our mentor for the project, who gave us a good guideline for the report throughout through numerous consultations. We would also like to extend our deepest gratitude towards everyone who have directly and indirectly helped us to complete our project.

Many people, especially our classmates, and team members themselves have made valuable comments and suggestions on this proposal which gave us an inspiration to improve our project. We thank all the people for their help directly and indirectly to complete our assignment.

We would also like to thank **DR. Anukruti Kaushal** for always being there to motivate us and enlighten us with his profound knowledge of the subject and suggesting us improvements to the projects. In addition, we would like to thank Department of Computer Engineering, Delhi Technological University for giving us the opportunity to work on this topic.

**CONTENT**

**1.INTRODUCTION**

1.1 What is library management system?

1.2 How is it developed.

**2.APPROACH**

2.1 How was the idea originated?

2.2 what were the reasons to think of this project

2.3 Prototype development

**3. PROTOTYPE Flowchart**

**4.Software tools used**

4.1 C++

4.2 MY SQL

**5.TESTING PHASE**

**6.Advantages of E-Library management system.**

**7.CODE**

**8.OUTPUT**

**9.References**

**INTRODUCTION**

Library management system in c programming.

The library management system is developed to make the basic library functions easier and much feasible.

According to various researches of universities it is found that the tasks and duties of library in normal mode is very time consuming and it is hard to keep the record of everything efficiently as it gets very cumbersome plus in a library it is very important to keep everything up to date which requires a lot of deletion and re-writing which is just not easily possible in regular library with a lot of sheets and papers.

Through this c-based program on library we aim to create much efficient and time saving software to manage all the administration functions of a library.



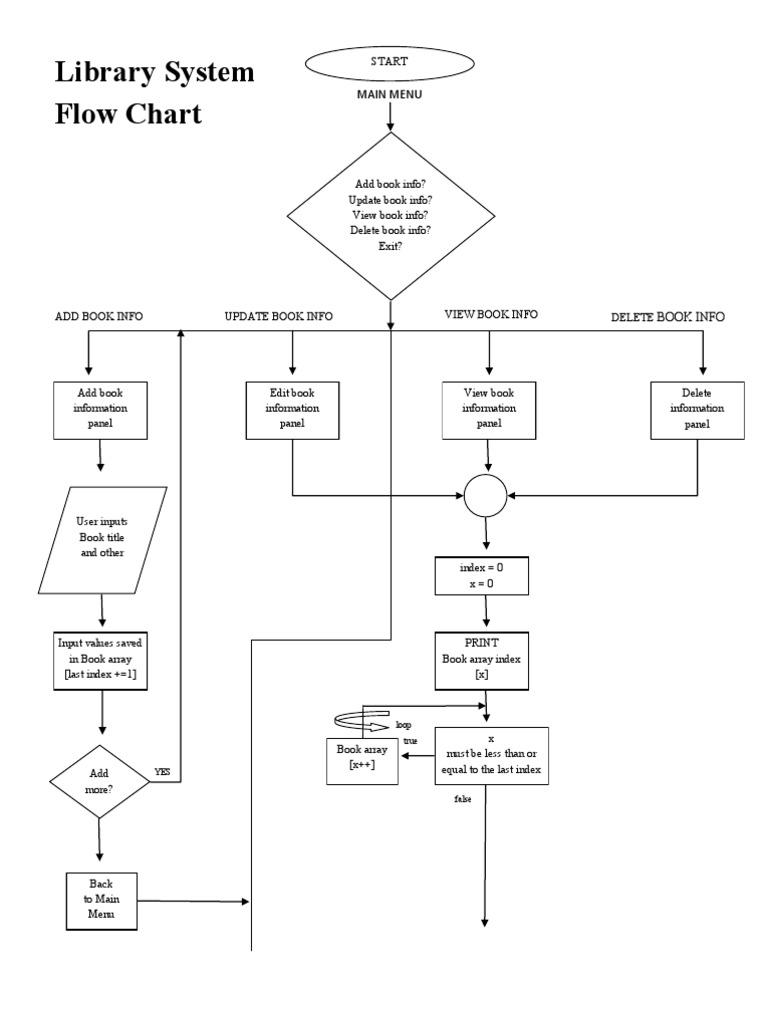
**APPROACH**

The approach was made in such a way that first we analysed the weaknesses of the existing library system and how to solve them. Some of the weaknesses that we found were:

Operating systems are vulnerable to human error. For instance, a librarian who misfiles a borrower's records or indexes a book incorrectly slows down the process and wastes students' time. Manual systems are also slow to operate. Instead of using a computer to issue and take back books, locating and updating a card index is slow and laborious. Manual systems are unable to store large amounts of data efficiently. With manual systems Librarian spend a lot of their time on mechanical, clerical tasks rather than liaising with library visitors.

Then in the next step we tried to originally think of how we can make some processes more efficient and then the idea of this project originated.

Next we developed a prototype of the system which is shown in the flowchart below



Then slowly overtime we started building our code blocks part by part to eventually end up with a big project in which we both contributed highly.

**SOFTWARE TOOLS USED:**

**C++**

**C++**  is a  general-purpose programming language  created by  Bjarne Stroustrup  as an extension of the  C programming language, or "C with  Classes". The language has expanded significantly over time, and modern C++ now has  object-oriented, [generic](https://en.wikipedia.org/wiki/Generic_programming), and  functional  features in addition to facilities for l ow-level  memory  manipulation. It is almost always implemented as a  compiled language, and many vendors provide C++ compilers, including the  Free Software Foundation, [LLVM](https://en.wikipedia.org/wiki/LLVM), [Microsoft](https://en.wikipedia.org/wiki/Microsoft), [Intel](https://en.wikipedia.org/wiki/Intel), [Oracle](https://en.wikipedia.org/wiki/Oracle_Developer_Studio), and IBM, so it is available on many platforms.[[9]](https://en.wikipedia.org/wiki/C%2B%2B" \l "cite_note-stroustruptcpppl-9)

C++ was designed with an orientation toward system programming and embedded, resource-constrained software and large systems, with performance, efficiency, and flexibility of use as its design highlights. C++ has also been found useful in many other contexts, with key strengths being software infrastructure and resource-constrained applications, including desktop applications, video games, servers (e.g. e-commerce, web search, or databases), and performance-critical applications (e.g. telephone switches or space probes).

C++ is standardized by the International Organization for Standardization (ISO), with the latest standard version ratified and published by ISO in December 2020 as *ISO/IEC 14882:2020* (informally known as C++20). The C++ programming language was initially standardized in 1998 as *ISO/IEC 14882:1998*, which was then amended by the C++03, C++11, C++14, and C++17 standards. The current C++20 standard supersedes these with new features and an enlarged standard library. Before the initial standardization in 1998, C++ was developed by Danish computer scientist Bjarne Stroustrup at Bell Labs since 1979 as an extension of the C language; he wanted an efficient and flexible language similar to C that also provided high-level features for program organization. Since 2012, C++ has been on a three-year release schedule with C++23 as the next planned standard.

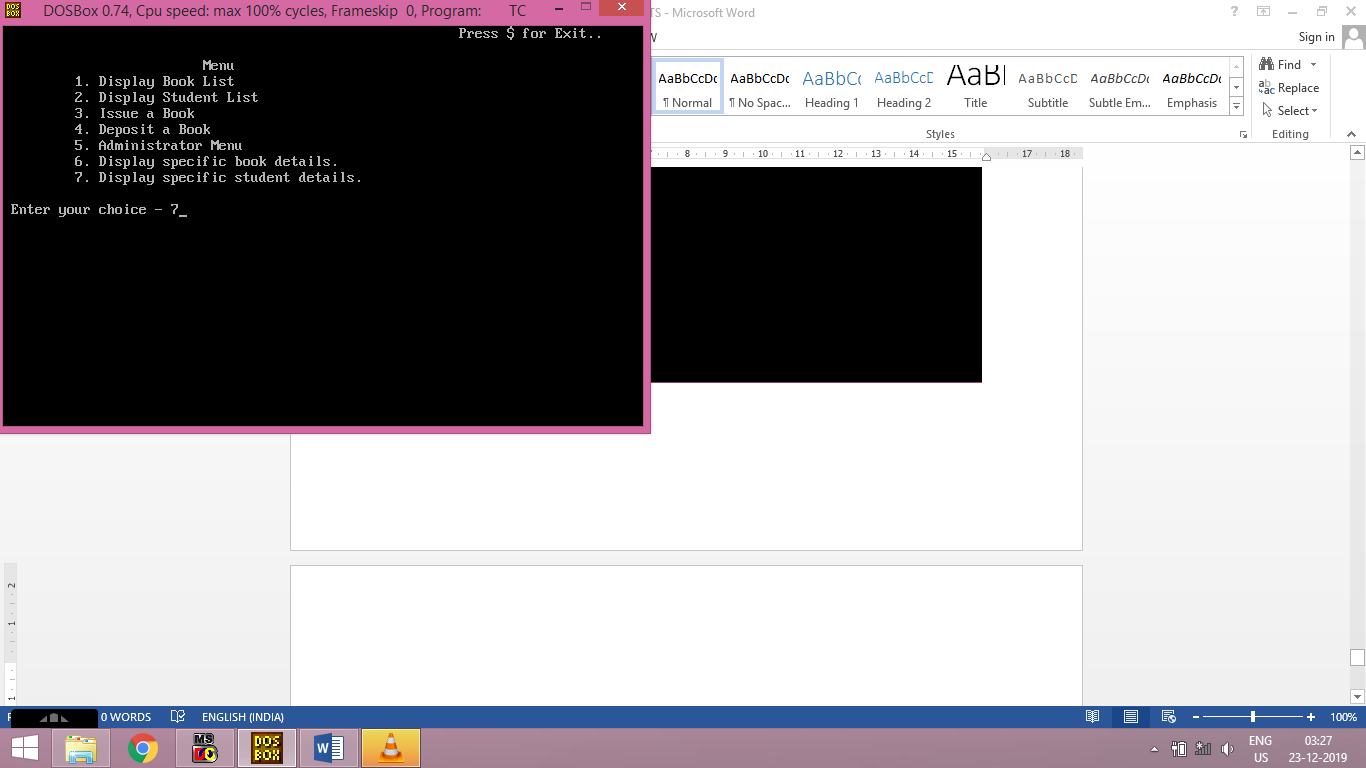
**SQL**

MYSQL- MySQL("My S-Q-L", officially, but also called "My Sequel") is (as ofJuly 2013) the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius daughter, My. The SQL phrase stands for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation .MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases

**TESTING**

After all this we started testing the programs for bugs and possible improvements in it to make it more efficient and better.

THE MENU



The menu was designed in such a way that every basic administration function needed in the library could be accessed through this place and makes it organized. Through menu we could display books and students list.

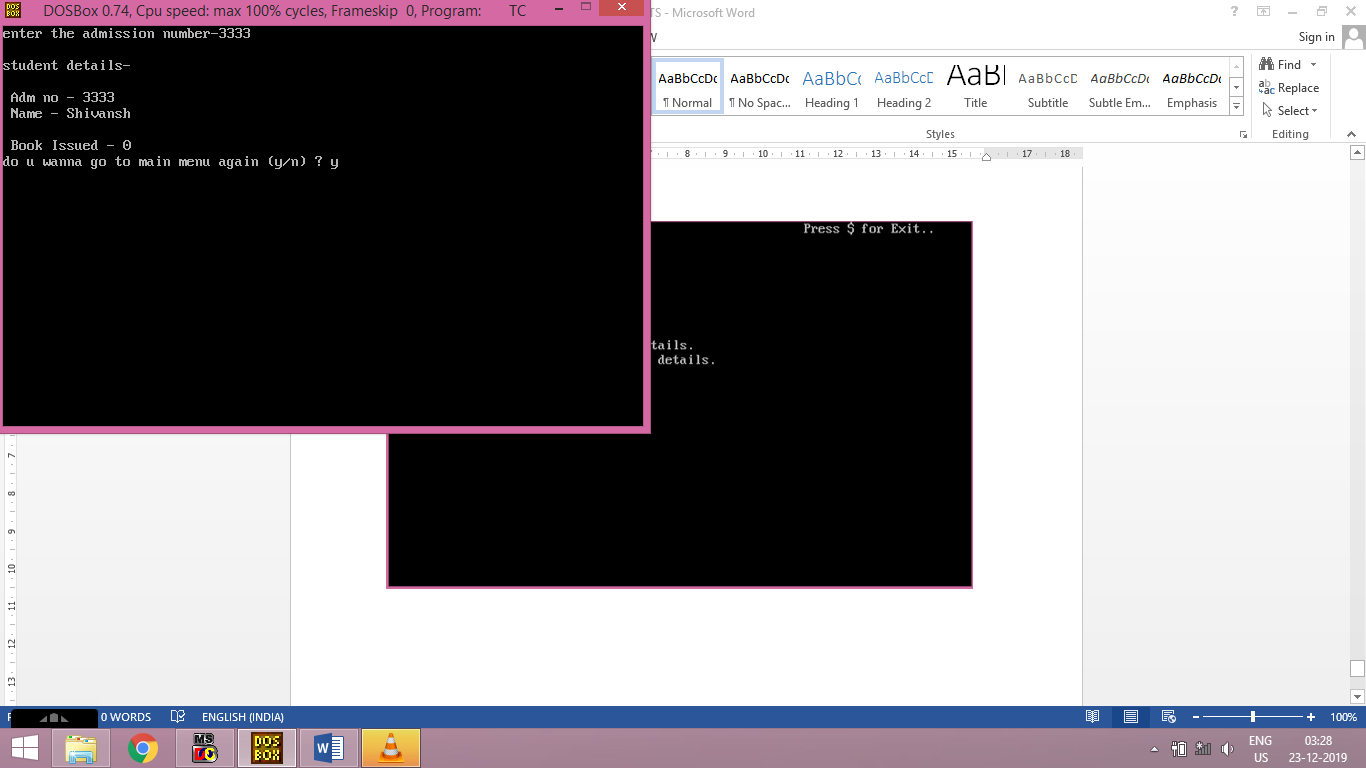
Issue a book

Deposit a book

go further into administrator menu

Display specific book details or specific student details

After this we needed our library data to be secure so no one other than admins could update data which is another advantage of this system over the manual one.  
So we made a security password system for admins if you need to upload data. This is more secure as shown below.



**Advantages of this program over manual library and its significance:**

* It is user-friendly software.
* It is cost-effective and easy to install.
* It helps in maintaining records.
* One can track any information through this system.
* It increases the efficiency.
* It saves human effort and time.
* It reduces the chances of error.
* It acts as an anti-theft.

Now as we have discussed all the features of the system let us see the code for the system:

**CODE**

// book library

#include<iostream.h>

#include<fstream.h>

#include<conio.h>

#include<stdio.h>

#include<string.h>

#include<process.h>

class book

{ char bkno[6];

char bkname[30],author[20];

public:

void add\_book()

{ cout<<"\n\t Enter Book Name- ";

gets(bkname);

cout<<"\n\t Enter Book no.- ";

gets(bkno);

cout<<"\n\t Enter Author- ";

gets(author);

cout<<endl;

}

void show\_book()

{ cout<<"\n Book No.- "<<bkno;

cout<<"\n Book Name - "<<bkname;

cout<<"\n Author - "<<author<<"\n";

}

void modify\_bk()

{

cout<<"\n book no. - "<<bkno<<"\n";

cout<<"\n modify book name - "<<"\n";

gets(bkname);

cout<<"\n modify author - "<<"\n";

gets(author);

}

void bkreport()

{

cout<<"\n\t"<<bkno<<"\t\t"<<bkname<<"\t\t"<<author;

cout<<endl;

}

char \* retbkno()

{ return bkno;

}

char \* retbkname()

{ return bkname;

}

} ; //class book ends here.....

class student

{ char stname[30], admno[6];

int check;

char stbno[6]; /\*to first check if book is

issued or not so that junk

value is not given in output

of book issued....\*/

public:

void add\_student()

{

cout<<" \n\t\t\t\tAdd New Student .." ;

cout<<"\n\nEnter Name - ";

gets(stname);

cout<<"\nEnter adm. no. - ";

gets(admno);

cout<<endl;

check=0;

}

void show\_student()

{ cout<<"\n Adm no - "<<admno;

cout<<"\n Name - "<<stname<<endl;

cout<<"\n Book Issued - "<<check;

}

void modify\_st()

{ cout<<"\n adm. no. - "<<admno<<endl;

cout<<"\nName - "<<stname<<endl;

cout<<"\n\nmodify name - ";

gets(stname);

}

int retcheck()

{ return check;

}

char \* retadmno()

{ return admno;

}

char \* retstname()

{

return stname;

}

char \* retstbno()

{

return stbno;

}

void addcheck()

{

check=1;

}

void resetcheck()

{

check=0;

}

void getstbno(char t[])

{

strcpy(stbno,t);

}

void streport()

{

cout<<endl;

cout<<admno<<"\t"<<stname<<"\t\t";

cout<<check<<endl;

}

}; //class student ends....

fstream f;

student s;

book b;

/\*\*\*\*\*\*\*\*\*\* function to write in file \*\*\*\*\*\*\*\*/

void write\_book()

{ clrscr();

char ch;

cout<<"\t\t\t\t << ADD NEW BOOK >>";

f.open("book.dat",ios::binary|ios::out|ios::app);

do

{ b.add\_book();

f.write((char\*)&b,sizeof(b));

cout<<"record added...\n\n";

cout<<"do you wanna continue to add records ? (y/n) ";

cin>>ch;

}

while(ch=='Y'||ch=='y');

f.close();

}

void write\_student()

{ clrscr();

char ch1;

cout<<"\n\t\t\t\t << ADD NEW STUDENT >>\n";

f.open("student.dat",ios::binary|ios::out|ios::app);

do

{ s.add\_student();

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

cout<<"Record added ...\n\n";

cout<<"do you wanna continue ? (y/n) ";

cin>>ch1;

}

while(ch1=='Y'||ch1=='y');

f.close();

}

//\*\*\*\*\*\*\*\*\*\* funtion to insert students from begining \*\*\*\*\*\*\*\*\*\*

void insert\_student()

{ clrscr();

char ch1;

cout<<"\n\t\t\t\t << ADD STUDENTS >>\n";

f.open("student.dat",ios::binary|ios::out);

do

{ s.add\_student();

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

cout<<"Record added ...\n";

cout<<"do you wanna continue ? (y/n) ";

cin>>ch1;

}

while(ch1=='Y'||ch1=='y');

f.close();

}

/\*\*\*\*\*\*\*\* function to search \*\*\*\*\*\*\*\*/

void book\_searchbyroll()

{

char ch2;

int f1=0;

cout<<"\t\t\t\t << SEARCH BOOK BY BOOK NO. >>";

f.open("book.dat",ios::binary|ios::in);

char roll1[6];

do{clrscr();

cout<<"\n Enter the book no to be searched - ";

gets(roll1);

while(f.read((char\*)&b,sizeof(b)))

{

if(strcmpi(roll1,b.retbkno())==0)

{

f1=1;

b.show\_book();

}

}

if(f1==0)

cout<<"\nBook not present...\n";

cout<<"\nDo you wanna continue (y/n) ? ";

cin>>ch2;

}

while(ch2=='y'||ch2=='Y');

f.close();

}

void student\_searchbyroll()

{ clrscr();

char ch3;

int f2=0;

char roll2[6];

cout<<"\t\t\t\t << SEARCH STUDENT BY ROLL NO. >>";

f.open("student.dat",ios::binary|ios::in);

do

{ clrscr();

cout<<"\nEnter the roll no. of student - ";

gets(roll2);

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

{

if(strcmpi(roll2,s.retadmno())==0)

{ f2=1;

s.show\_student();

}

}

if(f2==0)

cout<<"Student not found..." ;

cout<<"Do you wanna continue ? (y/n) ";

cin>>ch3;

}

while(ch3=='y'||ch3=='Y');

f.close();

}

void student\_searchbyname()

{ clrscr();

char ch3;

int f3=0;

char stuname[20];

cout<<"\t\t\t\t << SEARCH STUDENT BY ROLL NO. >>\n";

f.open("student.dat",ios::binary|ios::in);

do

{clrscr();

cout<<"enter the name of the student-";

gets(stuname);

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

{

if(strcmpi(stuname,s.retstname())==0)

{

f3=1;

s.show\_student();

}

}

if(f3==0)

cout<<"\nstudent not found\n";

cout<<"do you wanna continue ? (y/n)";

cin>>ch3;

}

while(ch3=='y'||ch3=='Y');

f.close();

}

void book\_searchbyname()

{ clrscr();

char ch4;

int f4;

cout<<"\t\t\t\t << SEARCH BOOK BY NAME >>\n\n";

f.open("book.dat",ios::binary|ios::in);

char bukname[30];

do

{clrscr();

cout<<"\tenter the name of the book - ";

gets(bukname);

while( f.read((char\*)&b,sizeof(b)))

{

if(strcmpi(bukname,b.retbkname())==0)

{

f4=1;

b.show\_book();

}

}

if(f4==0)

cout<<"\n\tbook not found...";

cout<<"\n\tdo you wanna continue ? (y/n)";

cin>>ch4;

}

while(ch4=='y'||ch4=='Y');

f.close();

}

//\*\*\*\*\*\*\*\*\* function to read specific record from file \*\*\*\*\*\*\*\*\*

void display\_bk()

{ clrscr();

char n[20];

cout<<"enter the book number-";

gets(n);

cout<<endl<<"book details-\n";

int flag=0;

f.open("book.dat",ios::binary|ios::in);

while(f.read((char\*)&b,sizeof(b)))

{ if(strcmpi(b.retbkno(),n)==0)

{ b.show\_book();

flag=1;

}

}

f.close();

if(flag==0)

cout<<"\n\nthe book does not exist";

getch();

}

void display\_studt()

{ clrscr();

char n[20];

cout<<"enter the admission number-";

gets(n);

cout<<endl<<"student details-\n";

int flag=0;

f.open("student.dat",ios::binary|ios::in);

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

{ if(strcmpi(n,s.retadmno())==0)

{ s.show\_student();

flag=1;

}

}

f.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\nenter the book number of the book-";

cin>>n;

f.open("book.dat",ios::binary|ios::in|ios::out);

while(f.read((char\*)&b,sizeof(b))&&found==0)

{ if(strcmpi(b.retbkno(),n)==0)

{ b.show\_book();

cout<<"\nenter the new details of the book\n";

b.modify\_bk();

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

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

f.write((char\*)&b,sizeof(b));

cout<<"\n\nrecord updated\n";

found=1;

}

}

f.close();

if(found==0)

cout<<"\n\nrecord not found\n";

getch();

}

void modify\_student()

{

char n[6];

int found=0;

clrscr();

cout<<"\n\n\tmodify student record....";

cout<<"\n\nenter the admission number of the student -";

cin>>n;

f.open("student.dat",ios::binary|ios::in|ios::out);

while(f.read((char\*)&s,sizeof(s))&&found==0)

{ if(strcmpi(s.retadmno(),n)==0)

{ s.show\_student();

cout<<"\n\nenter the new details of the student \n";

s.modify\_st();

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

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

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

cout<<"\n\nrecord updated\n";

found=1;

}

}

f.close();

if(found==0)

cout<<"\n\nrecord not found\n";

getch();

}

//\*\*\*\*\*\*\*\*\* function to delete record of file \*\*\*\*\*\*\*\*\*\*

void delete\_student()

{ clrscr();

char n[6];

int flag=0;

clrscr();

cout<<"\n\n\ndelete student...";

cout<<"\n\nenter the adm. no. of the student you want to delete";

cin>>n;

f.open("student.dat",ios::binary|ios::in|ios::out);

fstream f2;

f2.open("temp.dat",ios::binary|ios::out);

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

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

{ if(strcmpi(s.retadmno(),n)!=0)

f2.write((char\*)&s,sizeof(s));

else

flag=1;

}

f2.close();

f.close();

remove("student.dat");

rename("temp.dat","student.dat");

if(flag==1)

cout<<"\n\nrecord deleted";

else

cout<<"\n\nrecord not found";

getch();

}

void delete\_book()

{ clrscr();

char n[6];

int flag=0;

cout<<"\n\n\ndelete book...";

cout<<"\n\nenter the book no. of book you want to delete";

gets(n);

f.open("book.dat",ios::binary|ios::in|ios::out);

fstream f2;

f2.open("temp.dat",ios::binary|ios::out);

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

while(f.read((char\*)&b,sizeof(b)))

{ if(strcmpi(b.retbkno(),n)!=0)

f2.write((char\*)&b,sizeof(b));

else

flag=1;

}

f2.close();

f.close();

remove("book.dat");

rename("temp.dat","book.dat");

if(flag==1)

cout<<"\n\nrecord deleted";

else

cout<<"\n\nrecord not found";

getch();

}

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

function to display complete data

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

void display\_stlist()

{ clrscr();

f.open("student.dat",ios::binary|ios::in);

cout<<"\t\t\t\t << STUDENT LIST >>\n\n";

cout<<"Adm No \t\tStudent Name \t\t book issued\n " ;

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

{

s.streport();

}

f.close();

}

void display\_bklist()

{ clrscr();

f.open("book.dat",ios::binary|ios::in);

cout<<"\t\t\t\t << BOOK LIST >>\n\n";

cout<<"\tBook No \tBook Name \tBook Author \n\n " ;

while(f.read((char\*)&b,sizeof(b)))

{

b.bkreport();

}

f.close();

}

// \*\*\*\*\*\*\*\*\*\* function to issue book \*\*\*\*\*\*\*\*\*\*\*

void book\_issue()

{ fstream f1;

char sn[6],bn[6];

int found=0,flag=0;

clrscr();

cout<<"\n\n\t\t\t\tBOOK ISSUE ...";

cout<<"\n\n\tEnter The student's admission no.";

cin>>sn;

f.open("student.dat",ios::binary|ios::in|ios::out);

f1.open("book.dat",ios::binary|ios::in|ios::out);

while(f.read((char\*)&s,sizeof(s)) && found==0)

{

if(strcmpi(s.retadmno(),sn)==0)

{

found=1;

if(s.retcheck()==0)

{

cout<<"\n\n\tEnter the book no. ";

cin>>bn;

while(f1.read((char\*)&b,sizeof(b))&& flag==0)

{

if(strcmpi(b.retbkno(),bn)==0)

{

b.show\_book();

flag=1;

s.addcheck();

s.getstbno(b.retbkno());

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

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

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

cout<<"\n\n\t Book issued successfully \n\n";

cout<<"Please Note:: Fine will be imposed of";

cout<<"Rs 1 for each day after 10 days period. ";

}

}

if(flag==0)

cout<<"\n\nBook no. does not exist";

}

else

cout<<"\n\nYou have not returned the last book ";

}

}

if(found==0)

cout<<"\n\nStudent record not exist...";

getch();

f.close();

f1.close();

}

//\*\*\*\*\*\*\*\*\* function to deposit book \*\*\*\*\*\*\*\*\*\*

void book\_deposit()

{ fstream f1;

char sn[6],bn[6];

int found=0,flag=0,day,fine;

clrscr();

cout<<"\n\nBOOK DEPOSIT ...";

cout<<"\n\n\tEnter The students admission no. " ;

cin>>sn;

f.open("student.dat",ios::binary|ios::in|ios::out);

f1.open("book.dat",ios::binary|ios::in|ios::out);

while(f.read((char\*)&s,sizeof(s)) && found==0)

{

if(strcmpi(s.retadmno(),sn)==0)

{

found=1;

if(s.retcheck()==1)

{

while(f1.read((char\*)&b,sizeof(b))&& flag==0)

{

if(strcmpi(b.retbkno(),s.retstbno())==0)

{

b.show\_book();

flag=1;

cout<<"\n\nBook deposited in no. of days";

cin>>day;

if(day>10)

{

fine=(day-10)\*1;

cout<<"\n\nFine has to deposited Rs. "<<fine;

}

s.resetcheck();

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

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

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

cout<<"\n\n\t Book deposited successfully";

}

}

if(flag==0)

cout<<"Book does not exist";

}

else

cout<<"No book is issued..please check!!";

}

}

if(found==0)

cout<<"Student record does not exist...";

getch();

f.close();

f1.close();

}

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

administrator menu

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

void admin()

{

char c,ch ;

do{ clrscr();

cout<<"\t\t\t\t\t\t\t Enter $ for exit. ";

cout<<"\n\t 0. Add new book.";

cout<<"\n\t 1. Search book by book no.";

cout<<"\n\t 2. Search book by name .";

cout<<"\n\t 3. Delete book .";

cout<<"\n\t 4. Modify book details. \n\t";

cout<<"\n\t 5. Add new student.";

cout<<"\n\t 6. Search student by roll no. ";

cout<<"\n\t 7. Search student by name." ;

cout<<"\n\t 8. Delete student .";

cout<<"\n\t 9. Modify student details. ";

cout<<"\n\t #. add students from begining.";

cout<<"\n\nEnter your choice... ";

cin>>c;

cout<<endl;

switch(c)

{ case '0': write\_book();

break;

case '1': book\_searchbyroll();

break;

case '2': book\_searchbyname();

break;

case '3': delete\_book();

break;

case '4': modify\_book();

break;

case '5': write\_student();

break;

case '6': student\_searchbyroll();

break;

case '7': student\_searchbyname();

break;

case '8': delete\_student();

break;

case '9': modify\_student();

break;

case '#': insert\_student();

break;

case '$': exit(0);

default: cout<<"Invalid Input...";

break;

}

cout<<"\nDo you wanna go to Administrator menu again ? (y/n) ";

cin>>ch;

}

while(ch=='y'||ch=='Y');

getch();

}

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

password function for admin menu

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

void password()

{ clrscr();

char pass[20];

cout<<"\n\n\n\n\n\n\n\t\t\tENTER THE 4 DIGIT PASSWORD -\n\n\t\t\t\t";

gets(pass);

if(strcmpi(pass,"9876")==0)

admin();

else

cout<<"\n\n\n\t\t\twrong password entered\n\n\t\t\t";

getch();

}

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

Main Function

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

void main()

{

char c1,c2;

clrscr();

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

cout<<" \t\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ";

cout<<" \n\n\t\t\tWELCOME TO BOOK LIBRARY ";

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

getch();

do{

clrscr();

cout<<"\t\t\t\t\t\t\t Press $ for Exit..\n";

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

cout<<"\t 1. Display Book List ";

cout<<"\n\t 2. Display Student List ";

cout<<"\n\t 3. Issue a Book ";

cout<<"\n\t 4. Deposit a Book ";

cout<<"\n\t 5. Administrator Menu ";

cout<<"\n\t 6. Display specific book details.";

cout<<"\n\t 7. Display specific student details.";

cout<<"\n\n Enter your choice - ";

cin>>c1;

switch(c1)

{ case '1': display\_bklist();

break;

case '3': book\_issue();

break;

case '4': book\_deposit();

break;

case '5': password();

break;

case '6': display\_bk();

break;

case '2': display\_stlist();

break;

case '7': display\_studt();

break;

case '$': clrscr();

cout<<"\n\n\n\t\tEnd of program";

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

getch();

exit(0);

break;

default: cout<<"\nInvalid Choice..";

break;

}

cout<<"\ndo u wanna go to main menu again (y/n) ? ";

cin>>c2;

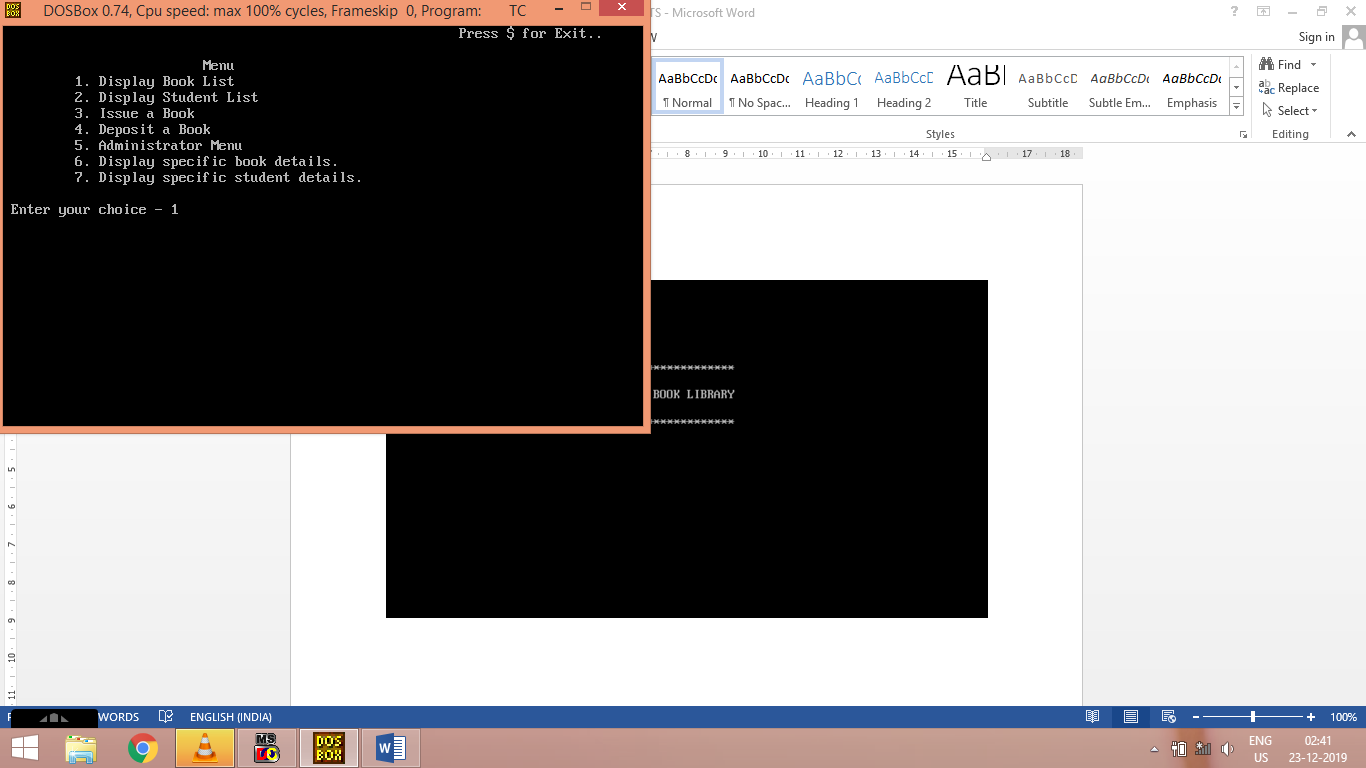
}while(c2=='y'||c2=='Y');

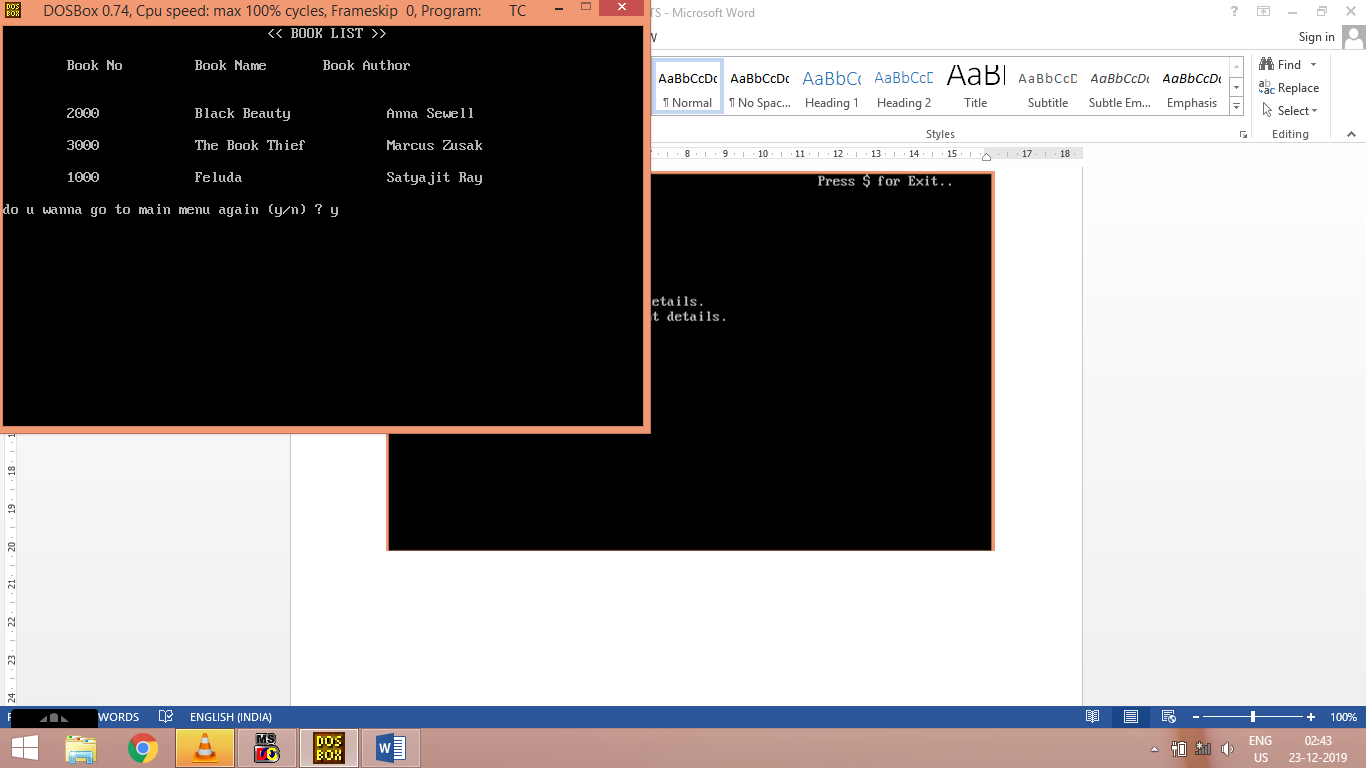
getch();

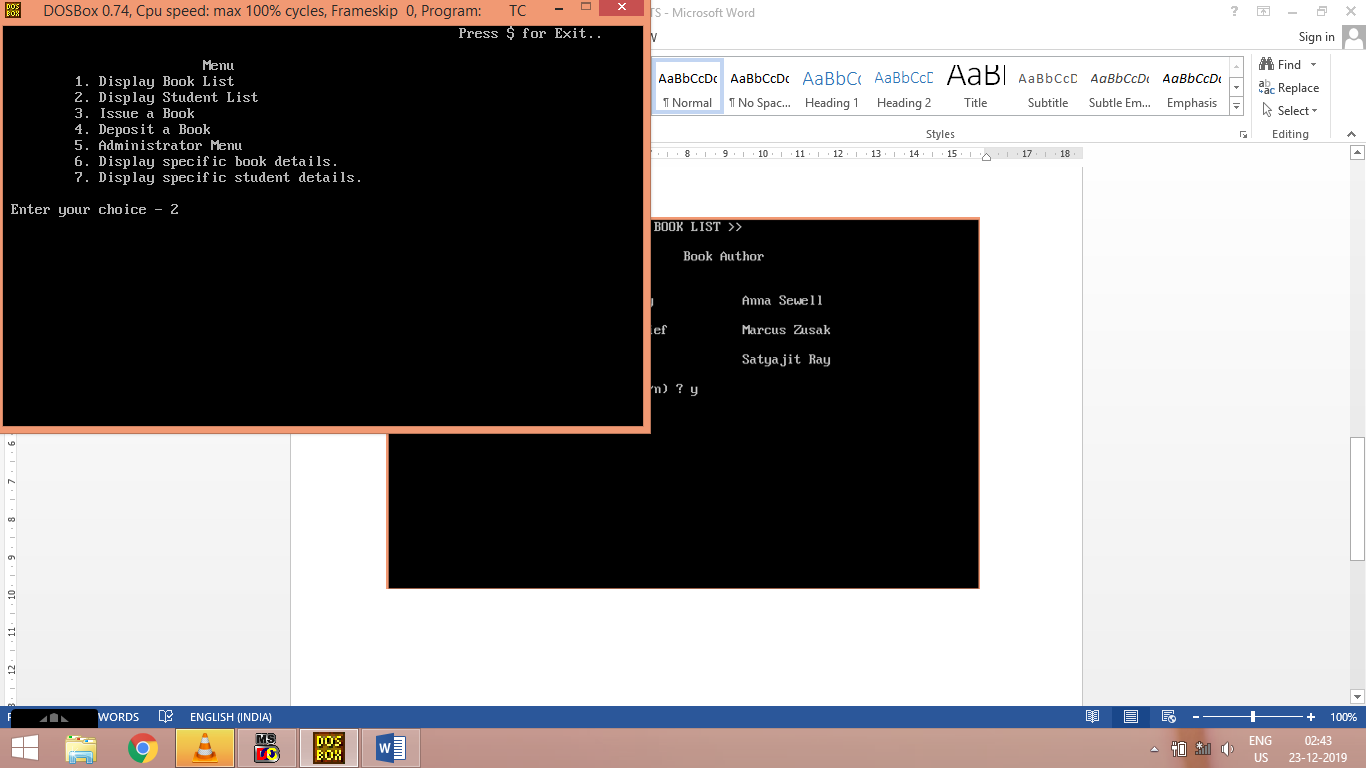
}

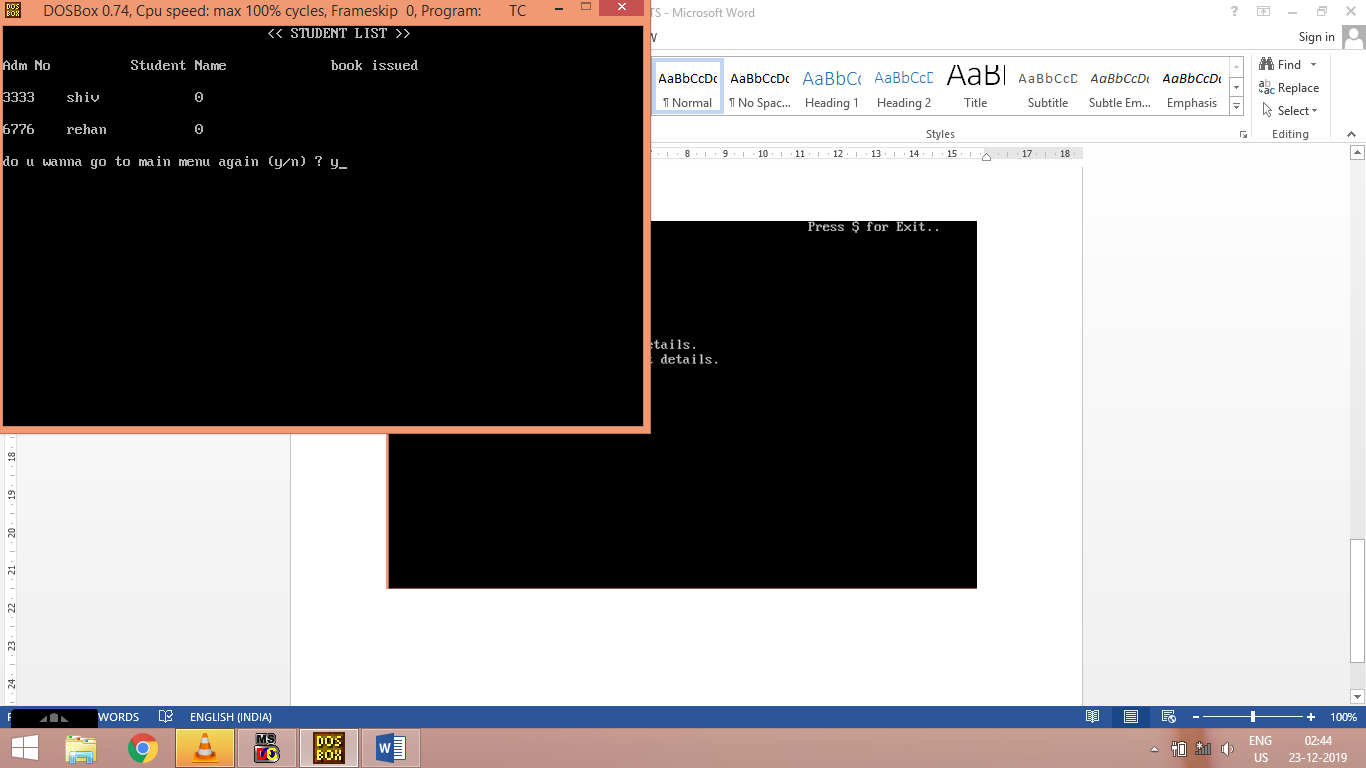
**OUTPUT**

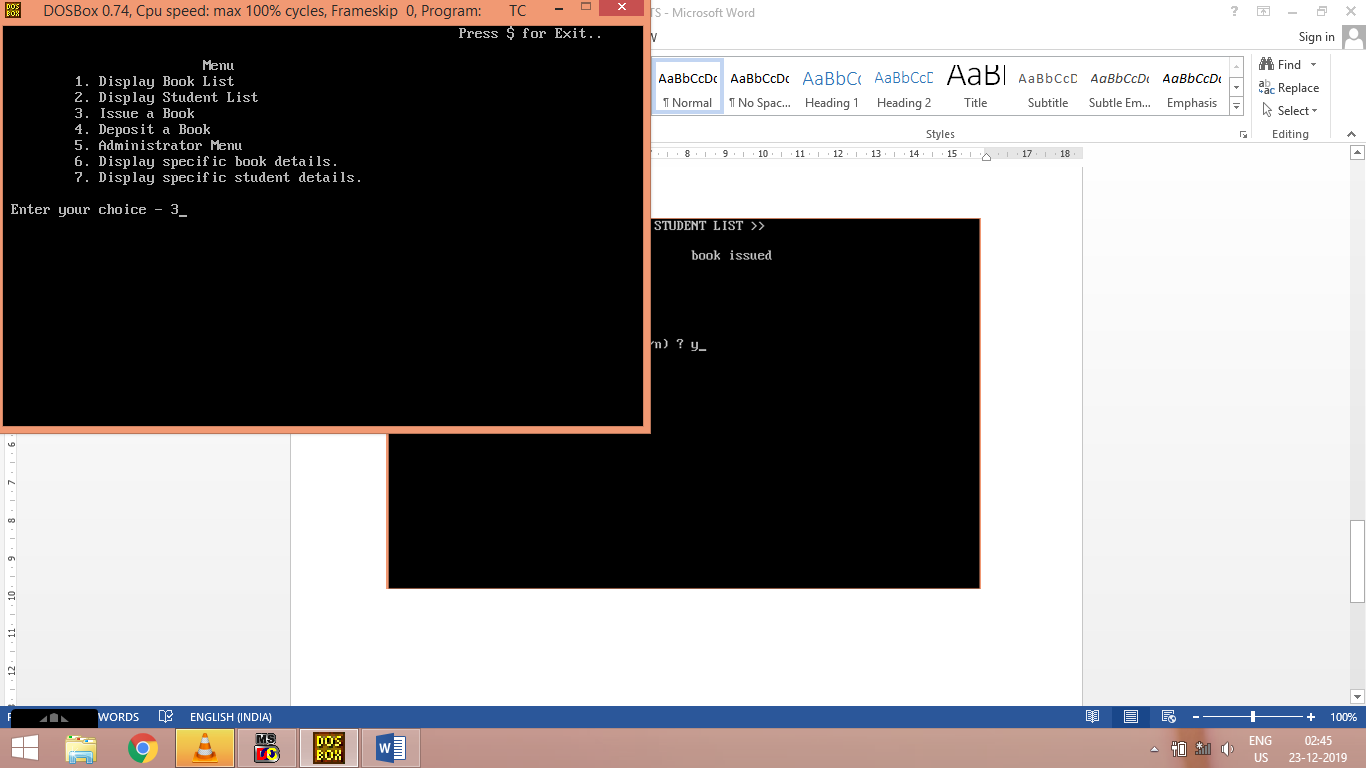
****

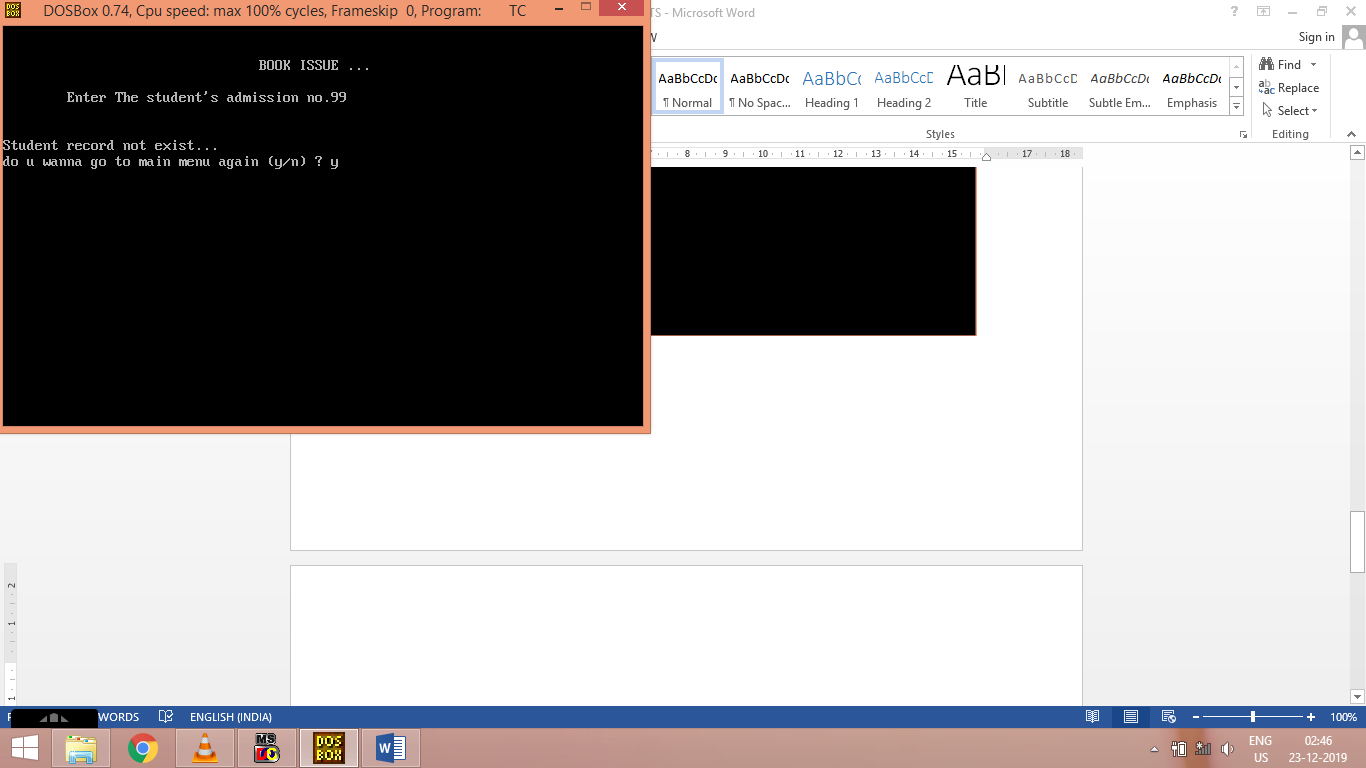


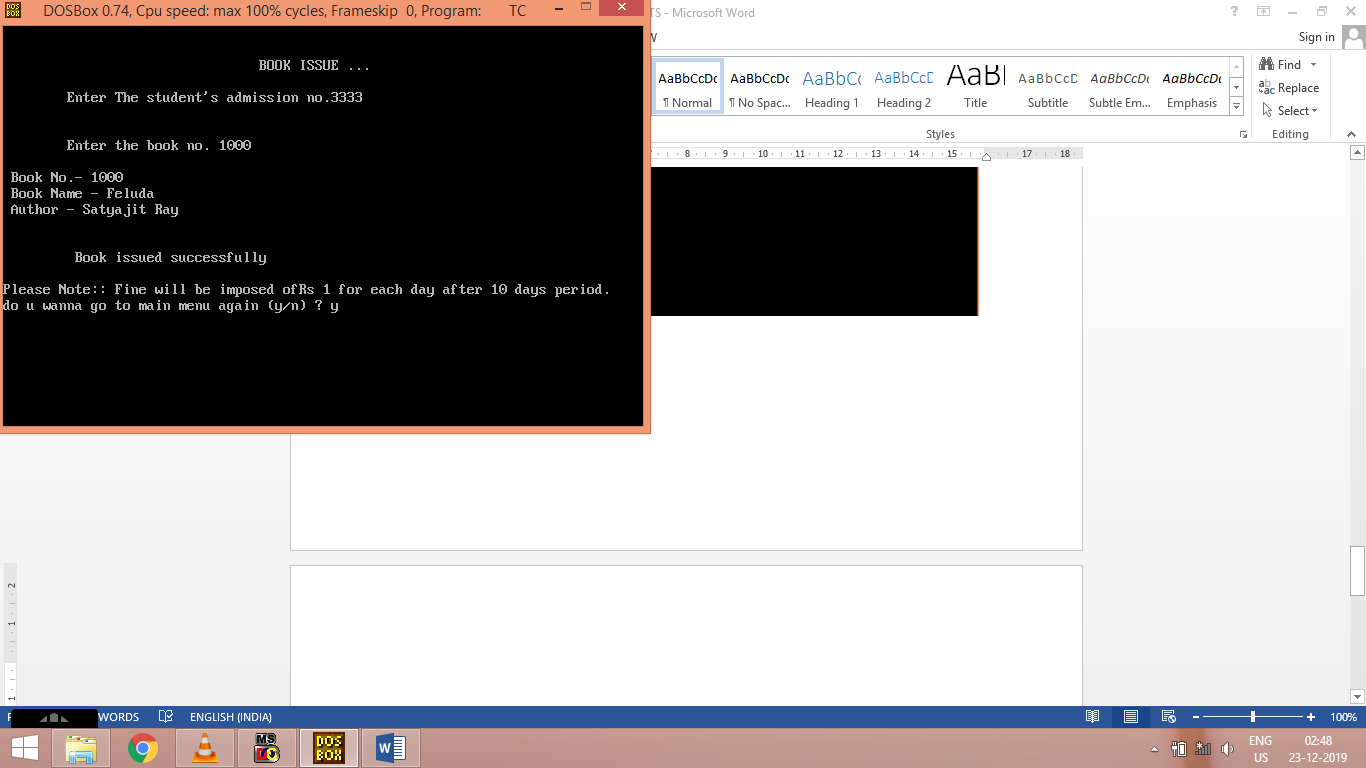


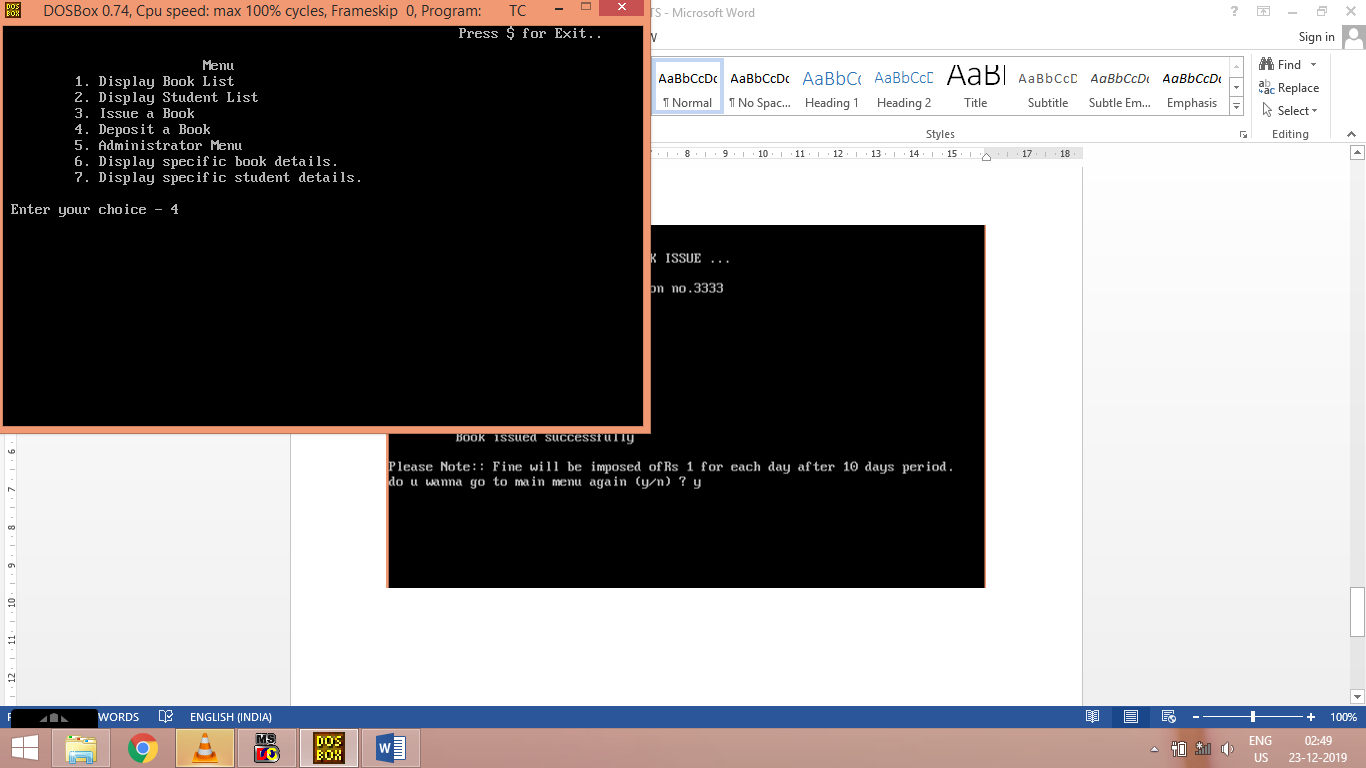


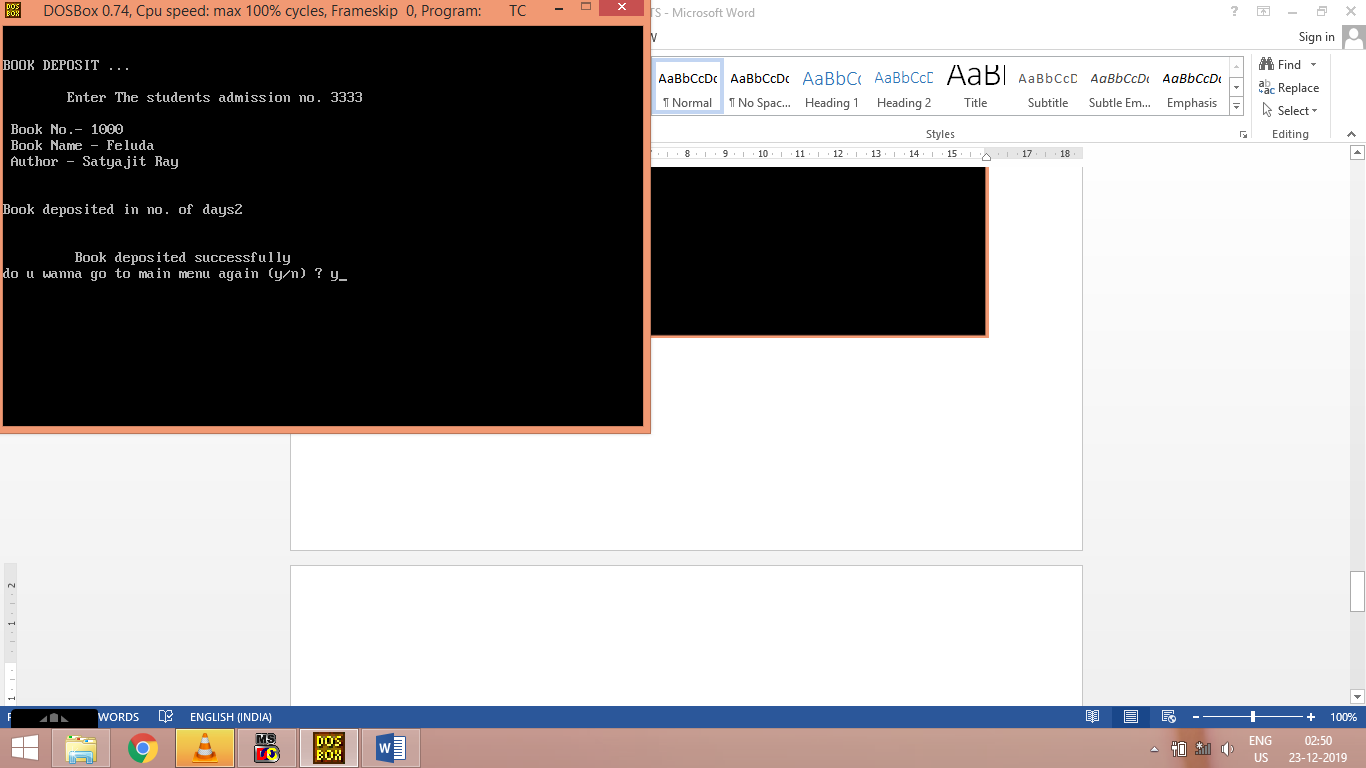


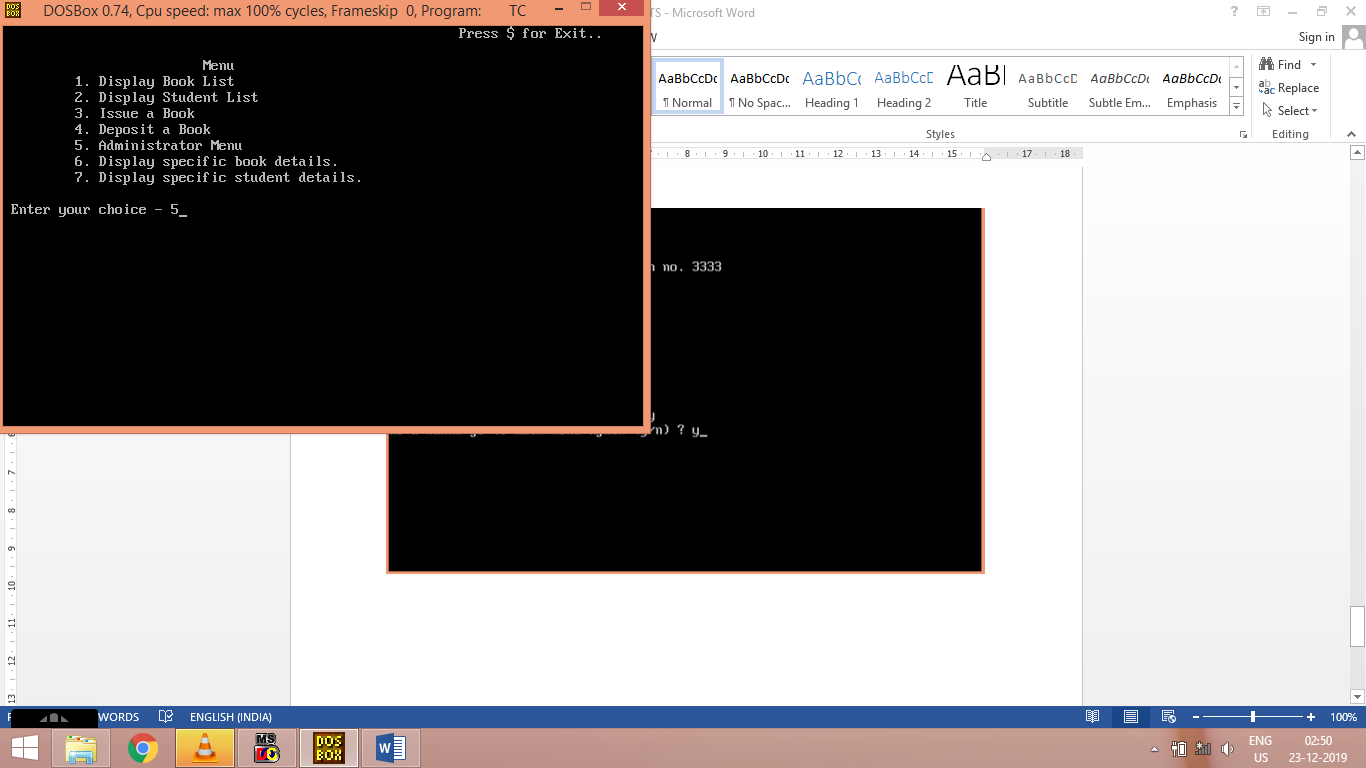


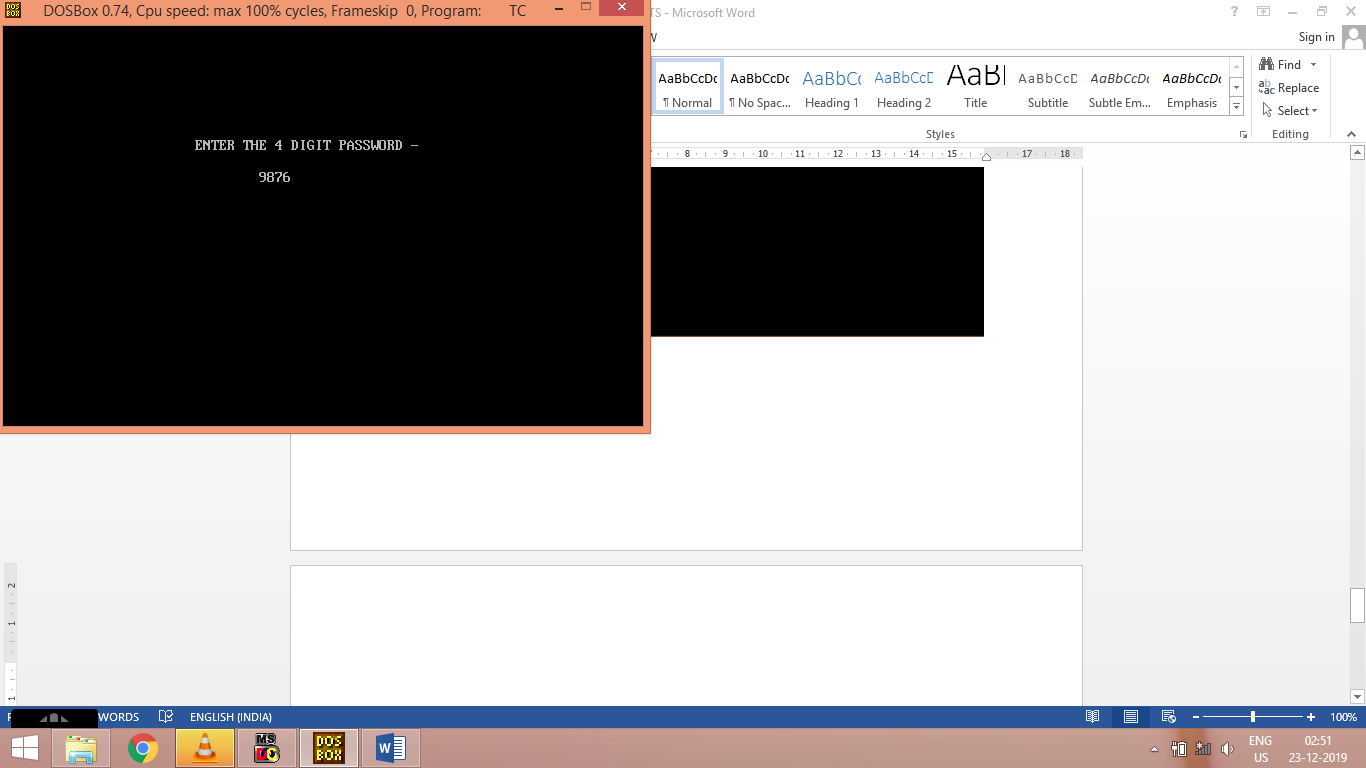


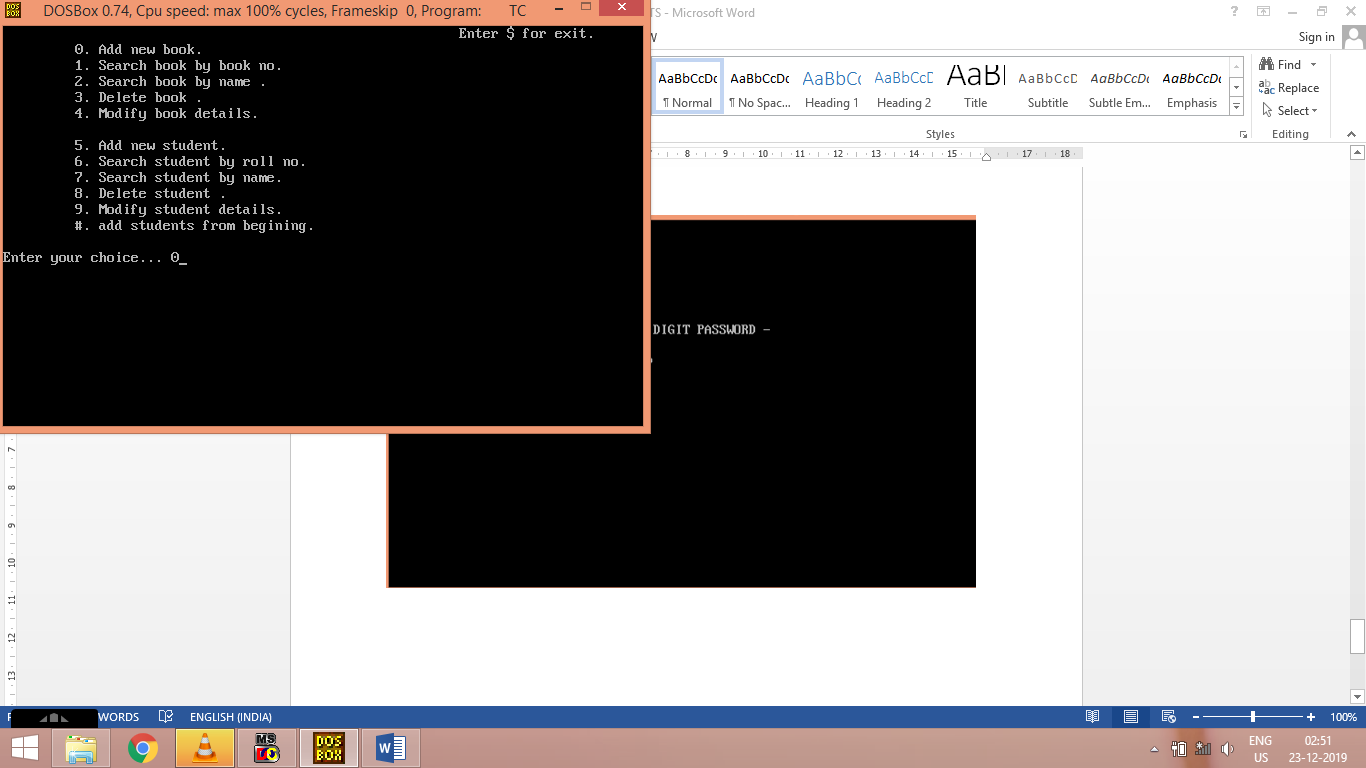


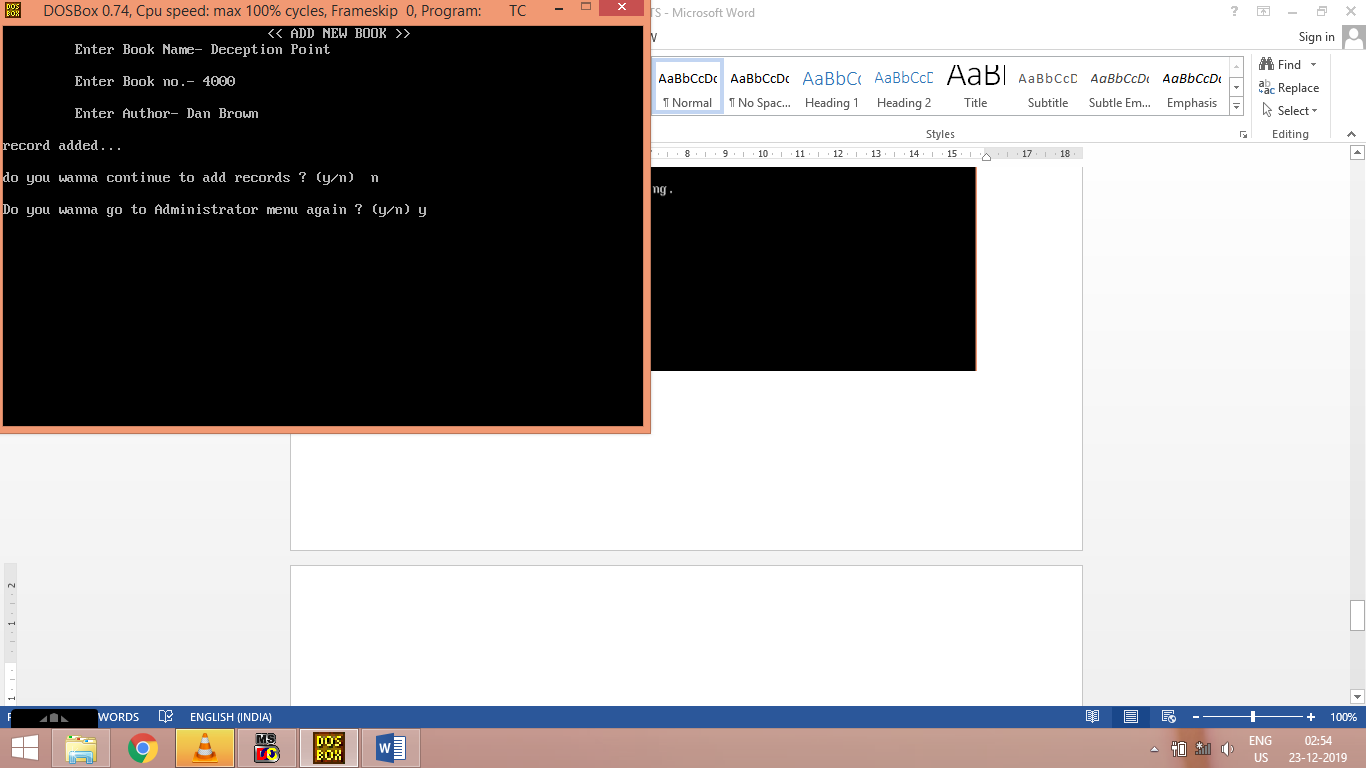


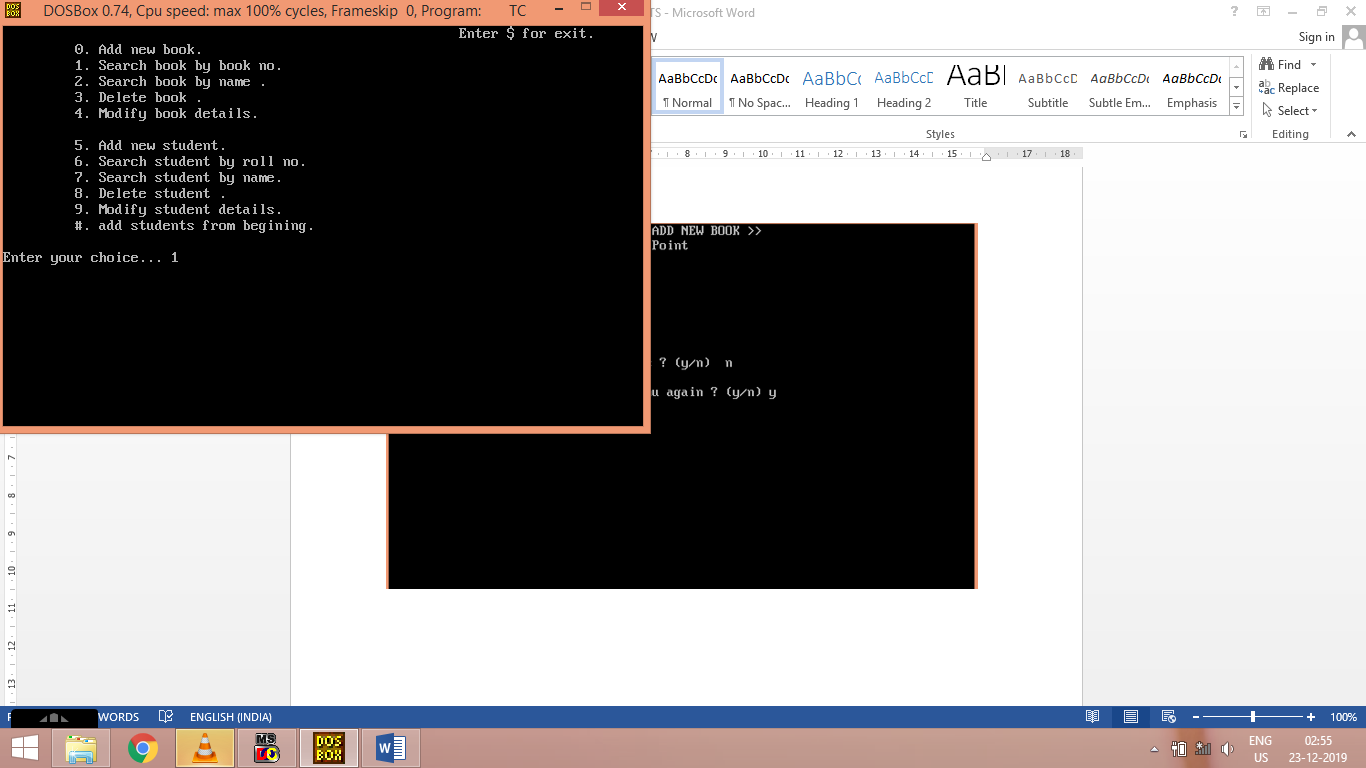


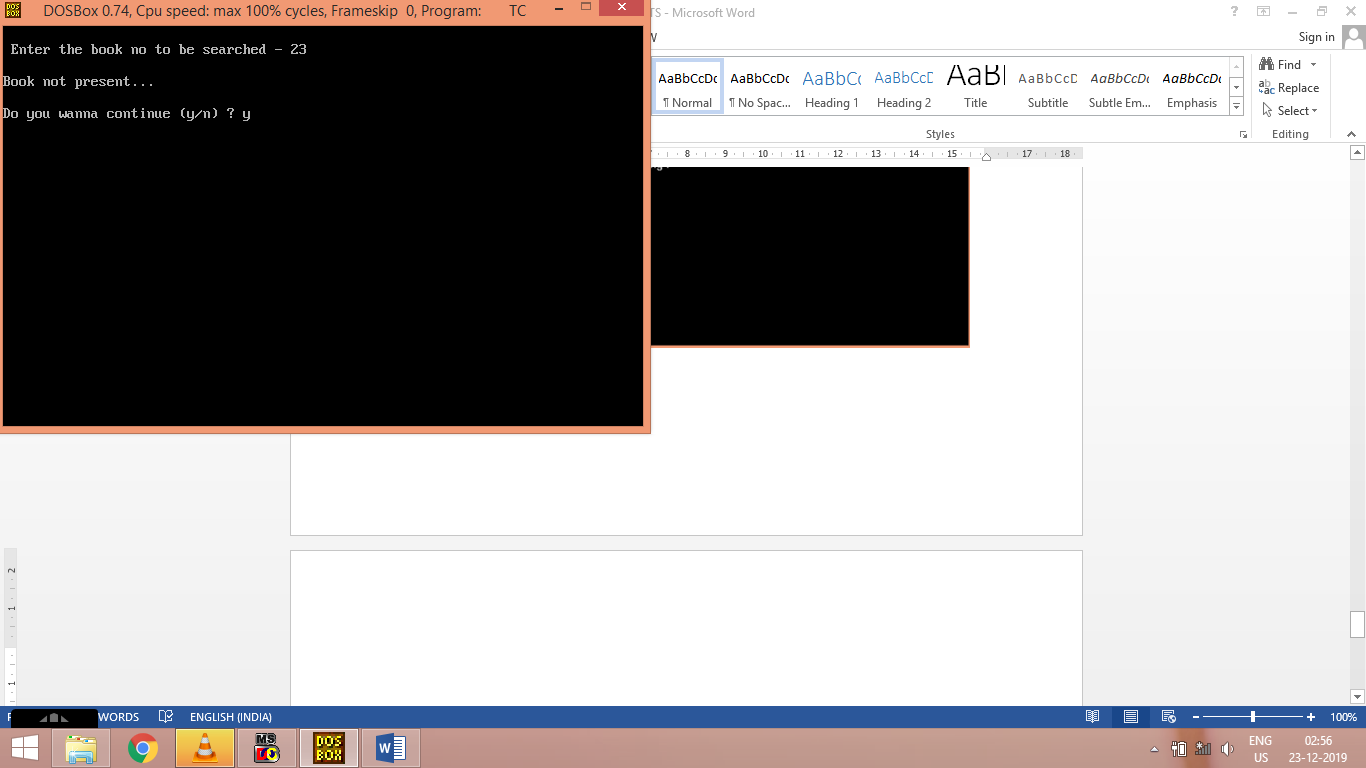


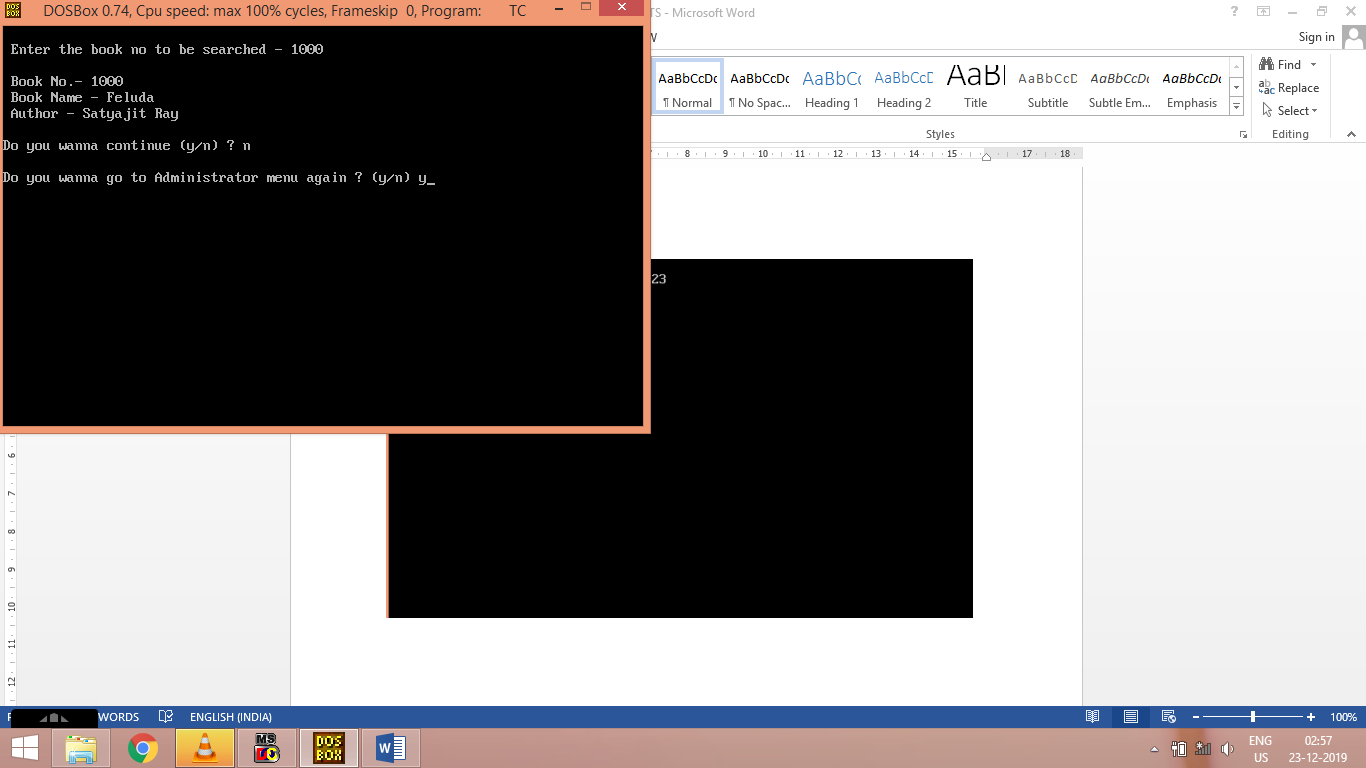


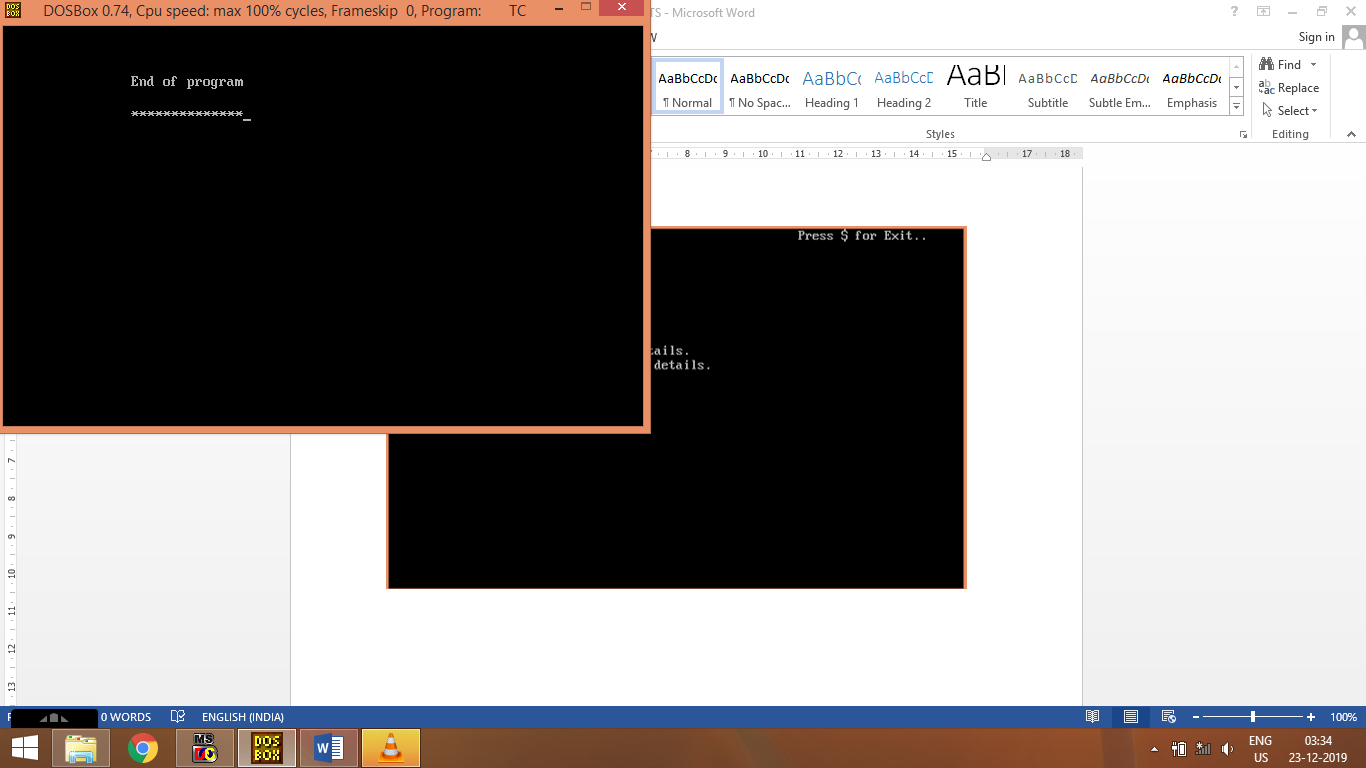












**REFERENCES**

<https://www.w3schools.com/html/html_intro.asp>

[https://www.udemy.com](https://www.udemy.com/)

<https://www.w3schools.com/mysql>

https://www.grin.com/document/205391