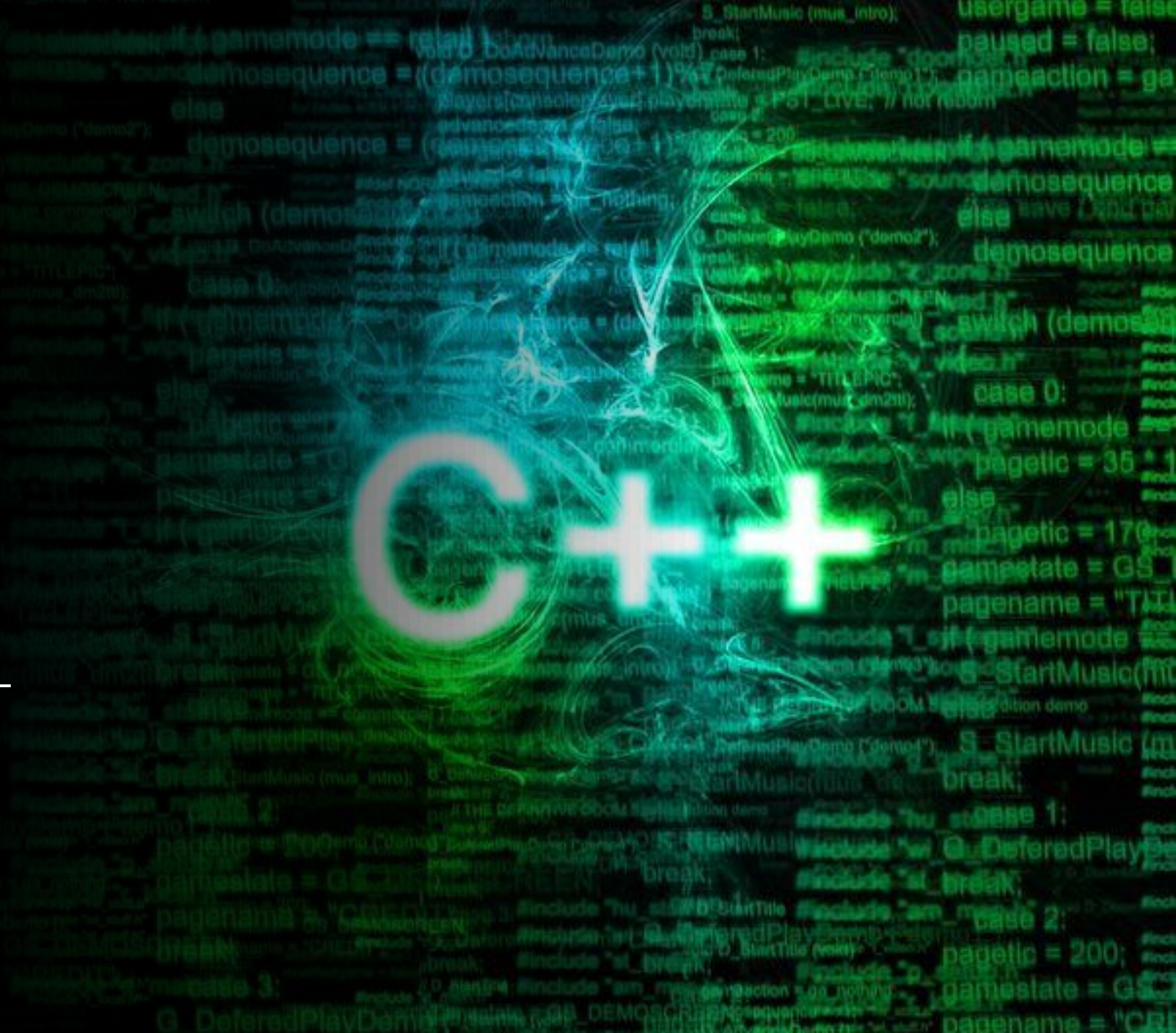
LIBRARY MANAGEMENT SYSTEM

PRESENTED BY

SINGAMSETTY CHANDUPRIYA

UNDER THE GUIDANCE OF (MENTOR)

AISHWARYA SAXENA



INTRODUCTION:

- Library Management System is based on a concept of recording owned books, issued books, returned books, students who have borrowed and many more.
- It is a software which handles the entire data of library. It makes the work of librarian very easy instead of writing data in a notebook. In past the librarians were using notebooks to write the data of books along with student's name who borrowed that book. So, it was very difficult to keep track on each book. If librarian wants to search for a particular book, then that task was very time consuming.
- So, to make this task easy we develop a console application in C++.

OBJECTIVE:

- In this Project we are required to develop a Library Management System as a console application in C++.

BACKGROUND:

- We have created separate logins for students and the librarian, in which the librarian is password protected.
- In this project, the librarian can add, update, delete and create books and can also assign the books to the students.
- The students can also view the list of the books available in the entire library database.
- The entire rights are given to the librarian to adding books, issuing books, and modify the book.
- This project uses file handling to store the data of books in a project.
- A Librarian can also be able to change the password.
- Reissuing and returning the books are the main features of this project.

HARDWARE AND SOFTWARE REQUIREMENTS:

HARDWARE TOOLS	MINIMUM REQUIREMENTS
Processor	i5 or above
Hard Disk	10GB
RAM	8GB
Monitor	17'' coloured
Mouse	Optical
Keyboard	122 Keys

SOFTWARE TOOLS	MINIMUM REQUIREMENTS
Platform	Windows, Linux or MacOS
Operating System	Windows, Linux or MacOS
Technology	C++
Scripting Language	C++
IDE	Code Blocks

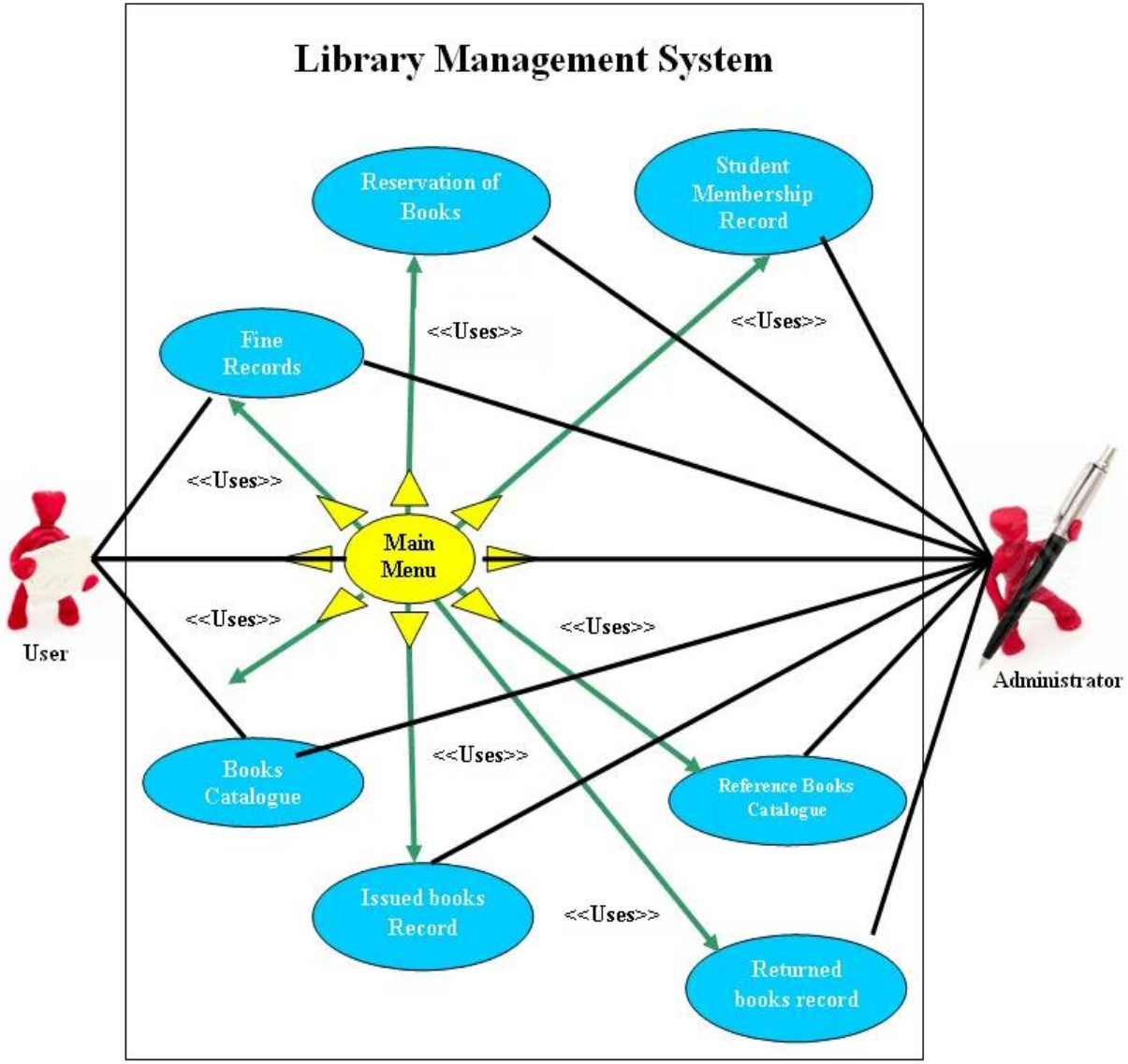
CODING:

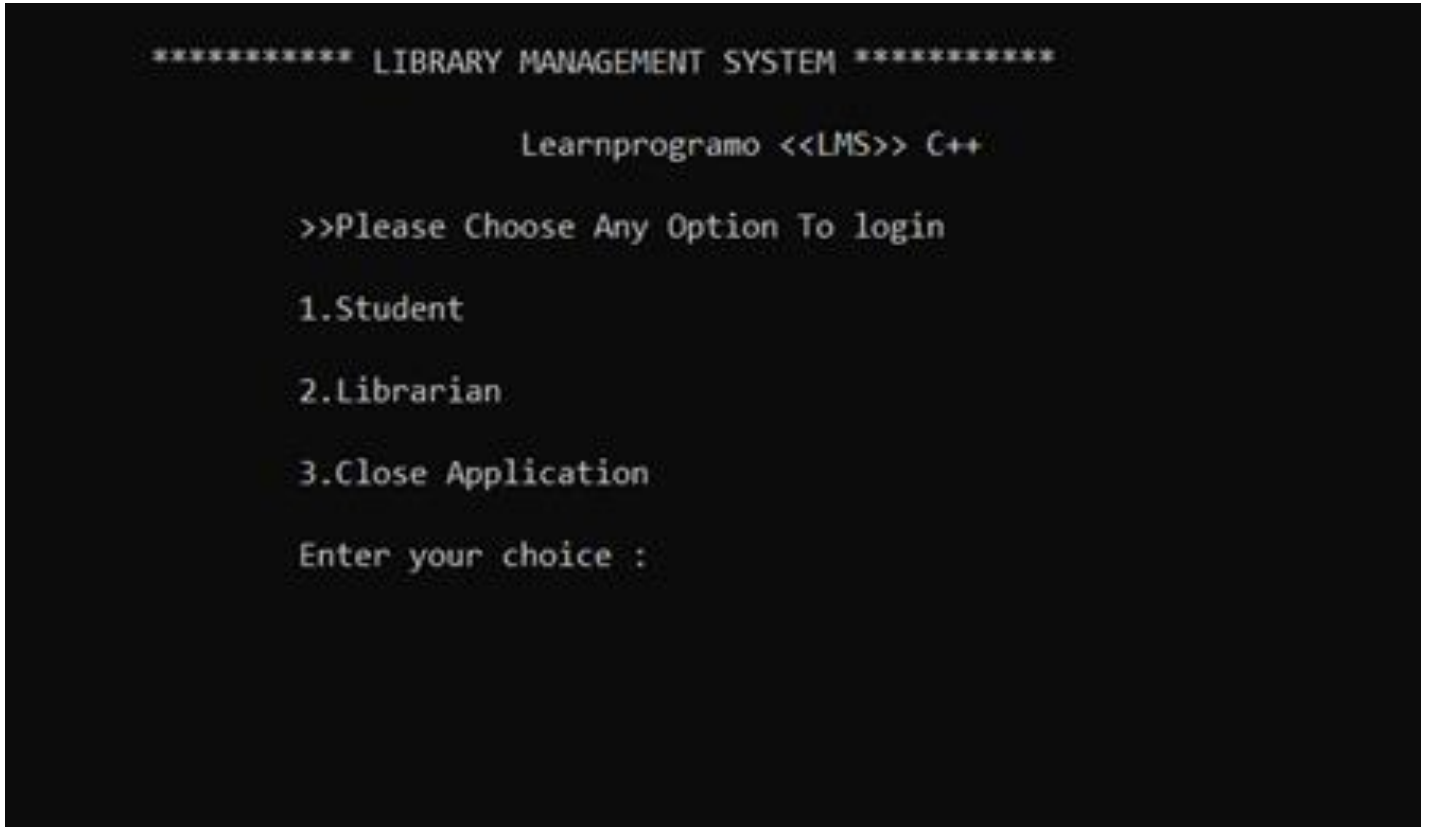
#include is a preprocessor directive that tells the preprocessor to include header files in the program. < >indicate the start and end of the file name to be included.

- **Iostream is** a header file that contains functions for input/output operations (cin and cout).
- **stdio.h** is needed for functions such as printf or fopen.
- **Stdlib.h** includes all standard library. Sometimes in some coding contests, when we have to save time while solving, then using this header file is helpful.

```
  1  #include<iostream>
  2  #include<stdio.h>
  3  #include<stdlib.h>
  4  #include<fstream>
  5  #include<string.h>
  6  #include<conio.h>
  7  using namespace std;
  8  class Lib
  9  {
 38  void Lib::getdata()
 39  {
 61  void Lib::show(int i)
 62  {
 73    void Lib::booklist(int i)
 74    {
112    void Lib::modify()
113    {
336    int Lib::branch(int x)
337    {
368    void Lib::see(int x)
369    {
453  void Lib::issue()
454  {
626  void Lib::fine(int d,int m,int y,int dd,int mm,int yy)
627  {
653  void Lib::der(char st[],int b,int x)
654  {
689  void Lib::get()
690  {
715  void Lib::student()
716  {
742  void Lib::pass()
743  {
784  void Lib::librarian()
785  {
814  void Lib::password()
815  {
907  int main()
908  {
909      Lib obj;
910      obj.get();
911      getch();
912      return 0;
913  }
```

- **fstream** object may be used to open a file for writing.
- **string.h** puts everything in the global namespace.
- **conio** means console input output.Some of its most commonly used functions are clrscr, getch, getche, kbhit etc. They can be used to clear screen, change color of text and background, move text, check whether a key is pressed or not and to perform other tasks.





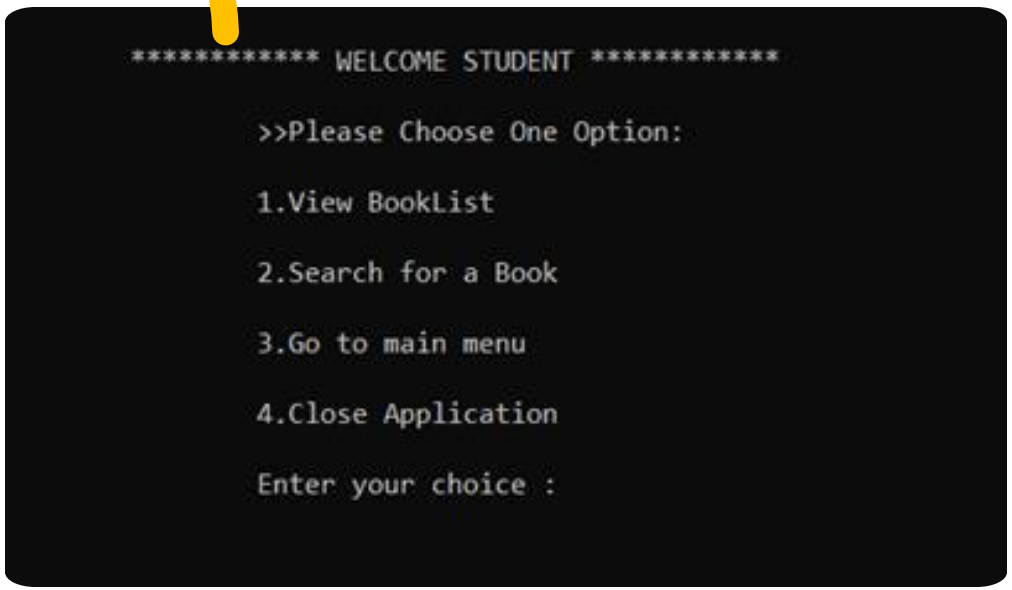
Main Screen:

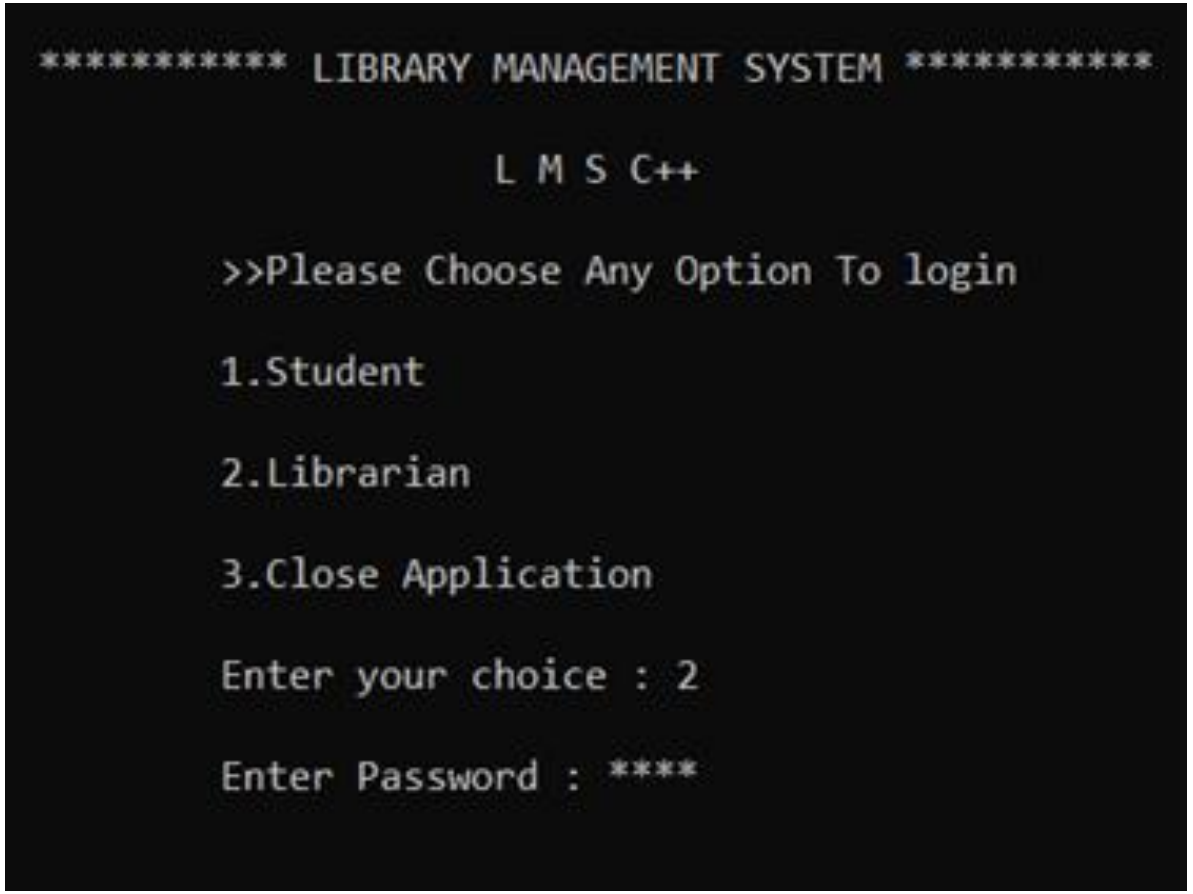
We have displayed the menu of Student, Librarian and close the application.

Student:

The student will not require additional sign in, he or she will be able to access the software directly.

- **1. View Booklist:** In this menu option all the students according to their branches will be able to view the books present in the database along with their details.
- **2. Search for a Book:** We have given access to the students to search for a particular book. The student can search book either by book name or by book id. Both the options are available in the project.
- **3. Go to Main Menu:** When the user has done the required operations and if he want to again move to the main menu, then pressing 3 as choice he'll moved to the main menu.
- **4. Close Application:** By pressing the choice as 4 the application will be closed.

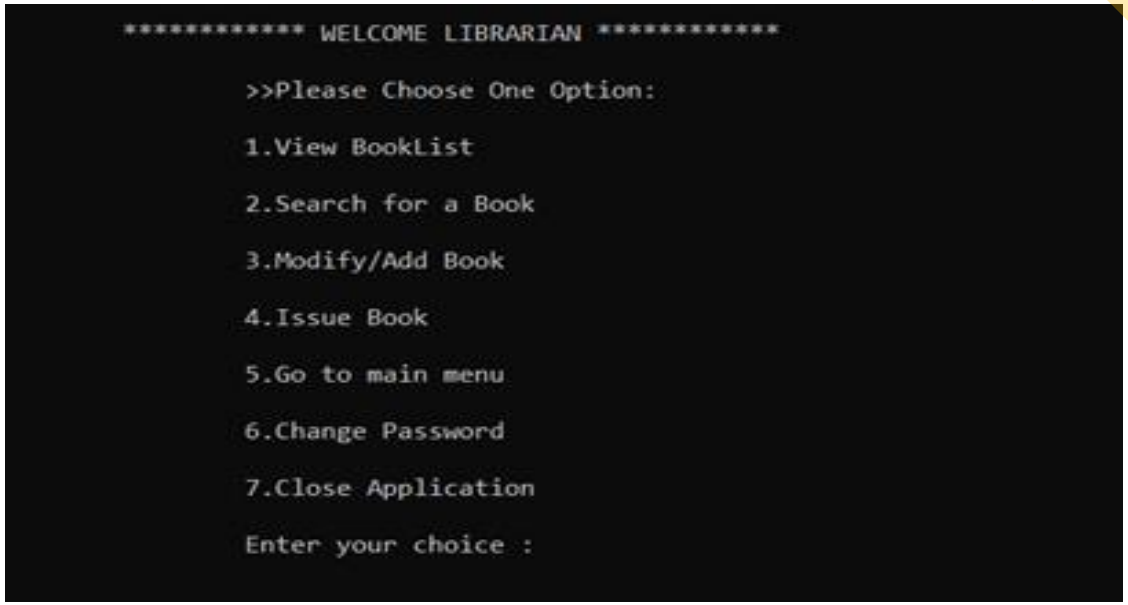




Librarian:

- To access the features of the librarian menu, He will require to sign in using the password which is "**pass**". We've also given the facility to change the password in the Librarian menu. Only Librarian has rights to change the password.
- If the password is incorrect the application will show the error of wrong password. And if the password is correct then the librarian menu will be visible to the user where he or she can do the operations displayed in the menu.

- **1. View Booklist:** Same as students view booklist, librarians will also be able to see the books available in the library database.
- **2. Search for a Book:** The Librarian can search book either by book name or by book id. Both the options are available in the project.
- **3. Modify/Add Book:** In this menu option Librarian can do three main operations i.e., Adding a Book, Deleting a Book and Modifying the existing Book.
- As we are using the file handling methods in this project, every time new file is generated to store the details of the books. i.e., **Booksdata.txt**.
- **4. Issue Book:** Due to this option the 70% of the work is been reduced. In this option Librarian can do the following operations:



- Issue a Book.
- View Issued Books.
- He can also search the students who issued the books.
- Librarian can also reissue the book to the same student.
- Return the Book
- To store the student details the separate file name **Student.txt** is been created.

FUTURE SCOPE:

This code can further develop and initialised in every library, and we can include many more components into the project that makes it more advantage to the librarians and it can set a due time to return a book and can remind to the librarian.

CONCLUSION:

In this Project we have developed a Library Management System as a console application in C++. I enjoyed building this project learnt new things and it helped improving my coding skills. By using different libraries to find local time and to manage time I successfully completed building this application.

REFERENCES AND BIBLIOGRAPHY:

1.Library Management Project in C++ (Computer Project) | ProjectAbstracts.com – Projects Ideas and Downloads (Reference Website)
2. Library Management System using C++ (sourcecodeera.com)(Reference code)