CS3101

Autumn 2021

(Project)

Designing a Library Management System in C
(Documentation)

Instructor : *Prof. Kripabandhu Ghosh*

Group : 22

Shivam Kumar (19MS123)
Swaraj Pradhan (19MS063)
Rishik Bhowmick (19MS064)
Adwait Naravane Bipinchanndra (19MS151)

Problem Statement

Library management system:

Design an interactive library system in C programming language. It should have two modes of access :

(i) **Admin** and (ii) **User**

There should be unique login ids for each. The system should ask for the login id and check it against a list. The admin and user ids for a person should be different. A person without a valid id should not be allowed to access the library. An Admin should typically be the librarian (or others responsible for the library). Users can be students or faculty members.

The allowed operations for each of these modes are:

Admin : Addition, deletion and updation of book records

User : Acquiring and returning of books

Each book should have a unique id, number of copies available, title, author, publisher etc. Each issue of a book should be recorded as well. The number of days for which a book can be borrowed should be dependent on the availability (more number of days for the books with a higher number of copies available). Also, there should be an upper limit on the number of books one can acquire at the same time. Books with less than 3 copies should be issued only to faculty members.

The system should have an interactive interface. It should show the current status of the availability and assign books accordingly. It should also take back books. The library database should be updated accordingly. It should also give the aforementioned rights to the admin.

In addition, a search facility should be present for both these modes allowing free-text search in general and also for specific fields of a book.

How to use ?

System Requirements : Nothing specific. Just a basic computer with a C compiler installed (MinGW gcc32 preferred). (for eg. 4 GB RAM, intel i3 or greater, OS: Windows, MacOs)

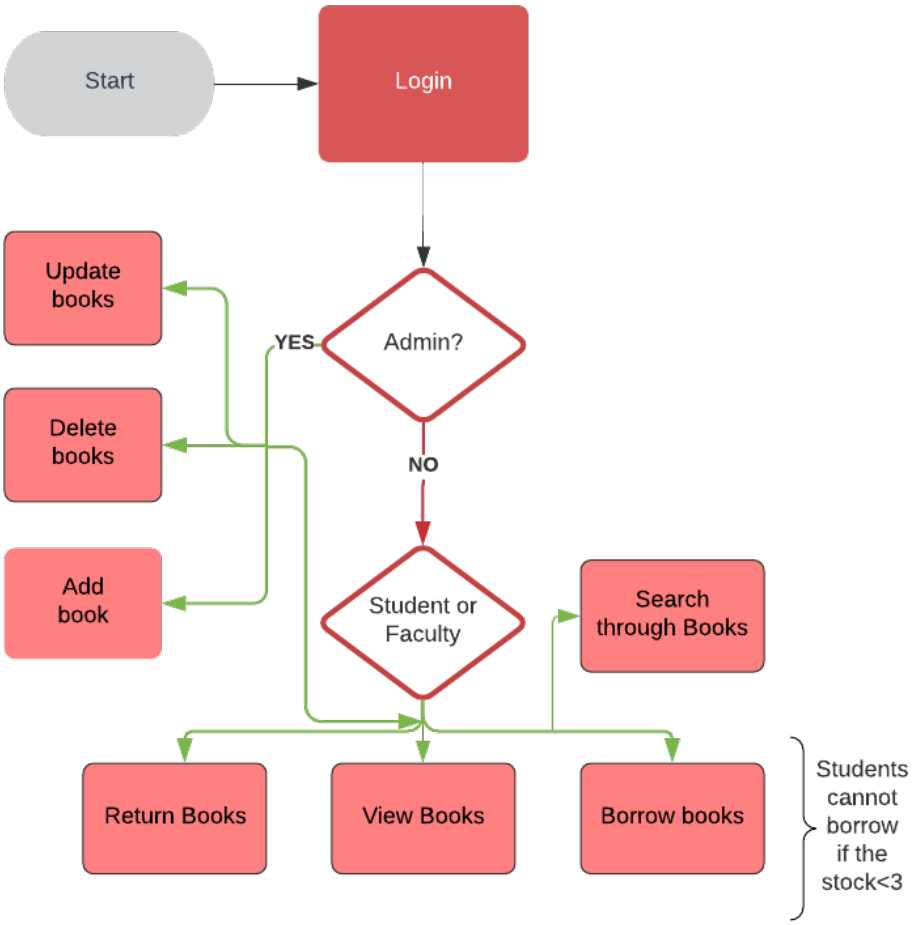
*Note : final code was tested successfully on :

System	:	Windows 10, 64-bit
Compiler	:	MinGW gcc32
Code Editor	:	Visual Studio Code

WARNING

DO NOT DELETE ANY FILES IN THE 'CS3101_LibraryManagementSystem-main' FOLDER.
All the files should be in the same directory for the code to run successfully.

Flow of program



Description of code

This is a basic library management program written in C language. The program is mainly based on file handling and usage of the composite data type **struct**. **Structure** is an user defined data type in C that allows us to combine data items of different kinds.Structures are used to represent a record that helps to keep track of the books in a library, user information etc. We have used four structures to keep track of **dates**, **user information**, **books information** and a **resgister for books borrowed and returned by the users**. The relevant data were written to 3 different files which contain user info, book info and book record. The text files "login_info.txt", "books_info.txt" and "register_book.txt" are used as our database.

Book info file :

Book_ID	Book_name	Author	Stock
INT PRIMARY KEY	VARCHAR[50]	VARCHAR[50]	INT

Login info file :

USER_ID	PASSWORD	USER_TYPE	MEMBER_TYPE
VARCHAR[30] PRIMARY KEY	VARCHAR[20]	VARCHAR[6]	VARCHAR[6]

Register file :

Book_ID	USER_ID	DATE_ISSUED	DATE_RETURN
INT FOREIGN KEY*	VARCHAR[30] FOREIGN KEY*	DATE	DATE

* The primary keys of 'Register file' and foreign keys of previous 2 structures.

The whole code is contained in a single file since different modules were coded and merged together. The modules are discussed below :

```
#include<stdio.h>                    //contains general functions like printf()
#include<string.h>                  //contains strcmp(),strcpy(),strlen(),etc
#include<ctype.h>                   //contains toupper(), tolower(),etc
#include<time.h>                    //contains time related functions
#include<stdlib.h>                  //standard functions & macros like NULL
```

1. INTERFACE AND main()

This part consists of the **login interface**, management of user info, designing of menus and calling all the functions in main().

void printMessageCenter(const char* message):

Prints the message at center of the screen.

void headMessage(const char *message):

Prints the heading message of the program “LIBRARY MANAGEMENT SYSTEM”

void welcomeMessage():

Prints “WELCOME” on the screen. Abd the user is required to enter any key to go to the next screen.

int isNameValid(const char *name):

Accepts names only with alphabets and spaces.

void menu():

Handles the menu. Prints the menu on the screen and uses
switch()
case:
to toogle and select items from the menu by calling the respective functions using the item number entered by the user and exits when the input is 0. It uses the user_type registered in the login screen to send the user to appropriate menu. Admins can add, delete and update book whereas other users can only view, search, borrow and return books.

void login():

This is the main function of the login interface. It asks the user for two variables viz. Username and Password. Then opens the login_info.txt file in read mode It then scans the files for matching username. If the username matches with the respective password it prompts “Login successful”, registers the user type (admin, user{stud, fac}) and sends the user to the menu screen by calling menu(). If username and password doesn’t match is invaid, it prompts “Login failed” and allows the user to try loging in 3 more times and exits the program after 3 failed attempts.

int main():

This is the driver function of the program. It calls the welcomeMessage() function and then the login() function.

2. ADMIN FUNCTIONS (Add, delete & update books)

This part is related to admin privileges and contains the functions which can be used by the admin-type users to add, delete and update books.

void headerAddBookPage():

Prompts that the user entered the “AddBook” page and asks to enter book details in the next screen.

int isBookPresent(int id):

Checks if the book with the entered id is present in the database i.e. the book_info file. Returns 1 id book is present and 2 if book is out of stock. (Also used as a helping function to the borrow book function)

void addBook():

Only admins can access this from the menu. It opens the book_info file in append mode (empty file doesn’t open and the program exits). Two integer variables viz. count and status to keep track of the status of adding books. After the header message, it prompts the user to enter the book_id then scans the id using isBookPresent() and stores the value returned in status. If status is 1 (book exists) and user is prompted to retry and count is increased. Else, count = 0. Same thing is done with bbok name, author name and stock. Count < 3 means that we have entered all values correctly and are valid. So, it’s appended into the book_info file. If input is invalid, user is prompted to try again. Else, file is closed and we go back to the menu.

void updateBook():

Opens the book_info file, asks for book id to update. If it doesn't exist, the admin is asked to retry. If it exists, admin is prompted to enter the new book name, author name & stock. We store them in some variables. A temporary file "temp" is created and the data of the book_info file is copied there except for the book id which is to be updated. Then the updated book data is written to the temp file. Then temp is renamed as book_info and the original book_info file is deleted. User is prompted if it's done successfully and sent back to the main menu.

void deleteBooks():

Deletes the specified book from the database. Opens the book_info file, checks for the book_id entered by user in the file. It makes a copy of the original file with all the data except for the data of the entered book_id. Then renames the copy file and deletes the original file.

3. USER FUNCTIONS (search and view books)

void searchBooks():

Opens the book_info file and prompts the user to enter the book name to search. If it matches with the book name in book_info file then it displays the whole information of the book. And if the book has less than 3 copies, it prints "not available forborrowing" if the user is a student. If book is not found it says 'no record". Then press any key to goback to man menu.

void viewBooks():

Reads the book_info file and prints the data of all the available books in database in formatted order.

4. USER FUNCTIONS (borrow and return books)

int isBookIssued(int book):

Check the register_book file and determines if the book is issued already. (used as a helping function for the borrow function).

void book_Issued(int book):

Decreases the stock of issued books in the book_info file.
(used as a helping function for the borrow function).

char* showBookName(int id):

Returns borrowed book name when the entered book_id matches with the one in the database.

void showBorrowedBooks(int list[], int *count):

Prints borrowed books on the screen with the help of showBookName(int id) function.

void borrowBook():

Allows the user to borrow books.
Asks the user to enter the book_id to borrow, check if it's available and if the user has already borrowed it once. If it's availabe for borrowing, the book details is printed on the screen and the user is asked to confirm borrowing (also checks if the book is less than 3 in quantity and if the user is a student, he is not allowed to borrow the book). If yes, then prints "book issued successfully", reduces the stock by 1 in the book_info file and the return date, writes the data of the borrower, borrowed book, issue and return date to the register_book file and back to the main menu. If book stock > 6, the user has to return it in 14 days and if stock > 12, he can borrow it for 21 days.

void returnBook():

Allows the user to return books.
Asks the user for book_id to return and if found in the borrowed books list, prompts the user to click "Y" if they want to return it. If yes, then increases the book stock by 1 in the database i.e. the book_info file. And then, we copy the content of register_book file to a temp file except the returned book, rename the temp file to register_book and delete the original file. Now the returned book is removed from the register and the user is prompted back to the main menu.

5. DATE AND TIME FUNCTIONS

Handles the dates of book issue and return.

int maxDateOfMonth(int m, int y):

Calculates the maximum number of days in a month.

Date returning_date(Date start, int period):

Calculates the return date of a borrowed book by taking the issue date and period to borrow as inputs.
(used in borrow function)

Date today_date():

Gets the current date from the system.
(used in borrow function)

LIMITATIONS OF THE CODE

- ✗ All files - books_info.txt, register_info.txt and login_info.txt exists and is not NULL.
- ✗ There are two user types - "admin" and "user", and further classified as "admin" as NONE, "user" as "fac"- for faculty members and "stud" - for students.
- ✗ If returning of book is after mentioned return date, fine is to be calculated by the librarian.
- ✗ Time operations are restriced to 28 days. (this fuction is called though)
- ✗ book_id s are to be entered by Admin only, so new numbers are generated by the Admin and put into the system.
- ✗ Names of books and authors take alphabets and blank spaces ONLY.
- ✗ It is restricted to one book per user. So if there are infinte books in library, then only the user can take infinte number of book.
- ✗ Only 100 books taken by the user will appear while returning the book.
- ✗ Search can be performed only by bookname and is case sensitive.

Library Management System (User Manual)

Welcome to the Library Management System. It is a small project that is being created using the C programming language. It will provide a computerized version of an interactive library database which will benefit the students as well as the staff/teachers. In this application, the user/admin can perform basic library management tasks like making a search, view the added books, check the number of copies available, returning/ borrowing a book etc.

Modules of Library Management System:

- Add Books
- Delete Books
- Search Books
- Issue Books
- Return Books
- Modify Books

Features of the Library Management System

The user manual describes the various aspects of the application. Depending on whether it's registered user or an admin, the following are the features-

- Registered users will be greeted with a login page in which both students and admin have to login with their unique username and password.
- Depending on whether it's a user or an admin, the level of accessibility will vary.
- A user will have access to limited features. A user can make an issue, depending on the number of copies available or return a book within a stipulated time.
- A user can also view the list of the books available in the entire library database.
- A user can be both student and faculty.
- A student won't be allowed to issue a book, if it has less than three copies available.
- There is a maximum limit to the number of books that can be issued.
- The entire rights and permission are given to an admin. An admin has access to features like adding, deleting, issuing or modifying any book.
- The number of days for which a book can be issued depends on the number of copies available.

Working of the application

In the zip folder, there is the executable.exe file where the entire project can be run manually. Here, the working of the entire application with explanation is provided.

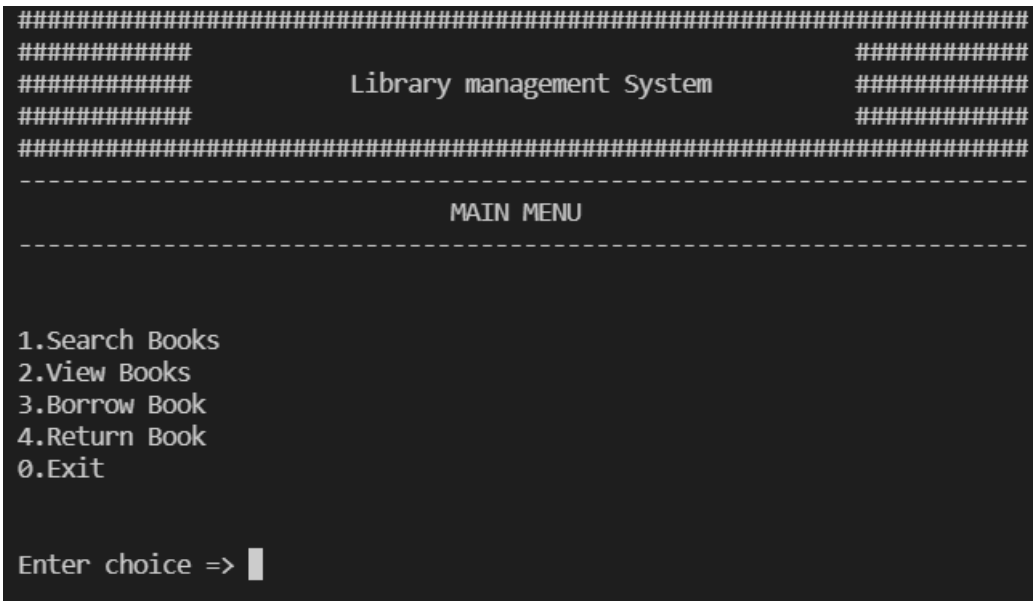
Main Screen:

1. When the program is being run from the compiler, the following welcome message is displayed in the terminal.

2. Press any key to proceed further.

3. In the Login page, enter the Username and Password which is registered in the database. The Username and Password are case sensitive.

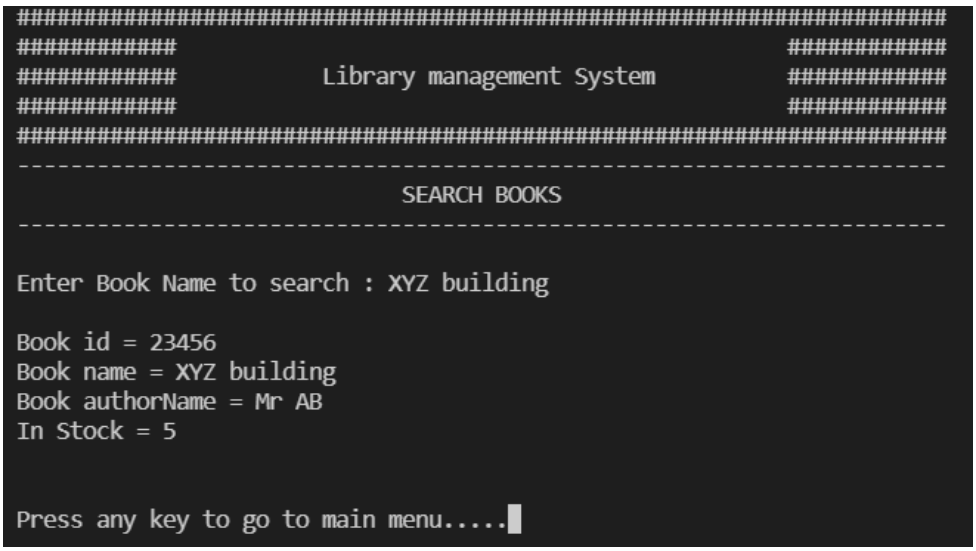
4. Now, depending on whether the registered user is an admin, the Main Menu will appear accordingly.
    - If he/she is a normal user, then the following options will appear in the Main Menu:



On logging in as an admin, following menu will be visible in the terminal:

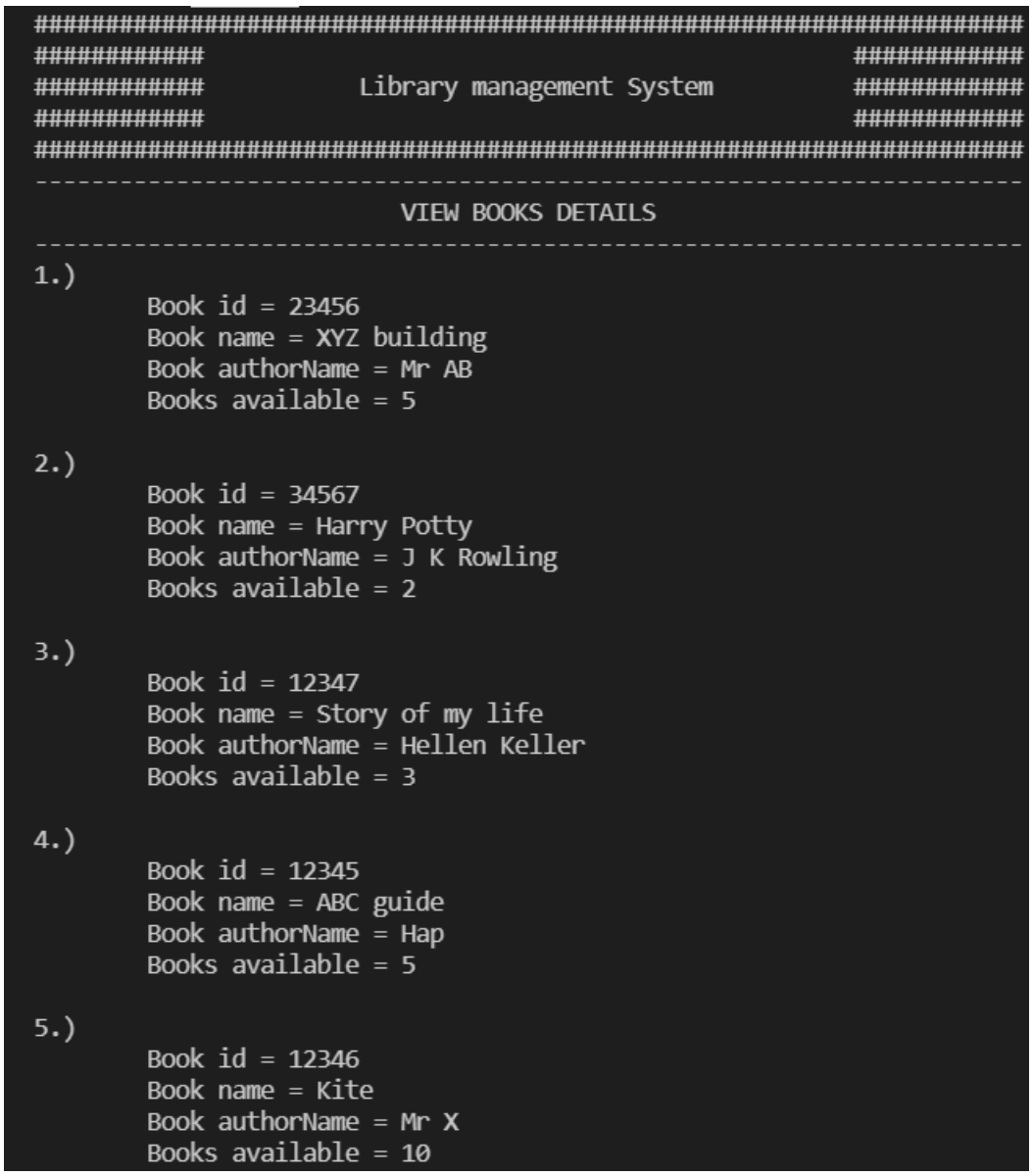
1. Search Books

Through this option, the user can get access to any particular book. The search can be made by typing in the name of the book. If the required book is available in the database, then the program will display the book id along with the author name and the number of books in stock. The name of the book is case sensitive.



2. View Books

In this menu option, the user will be able to view the entire book collection that is present in the library database along with their details.

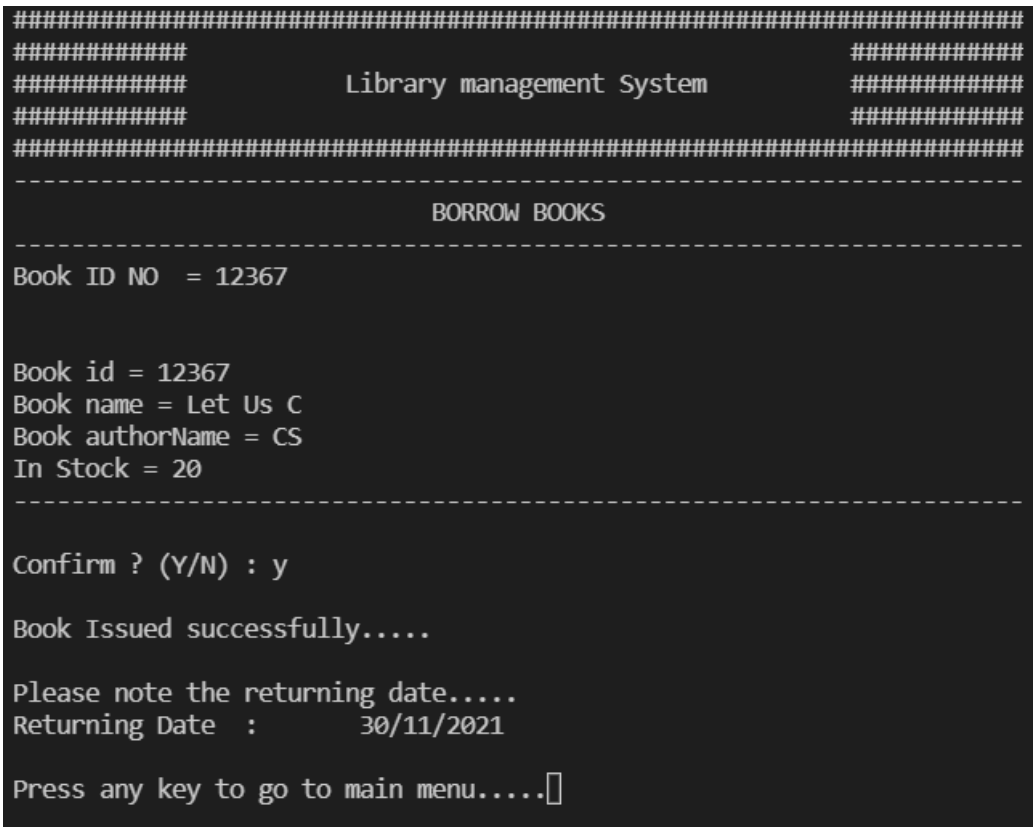


3. Borrow Book

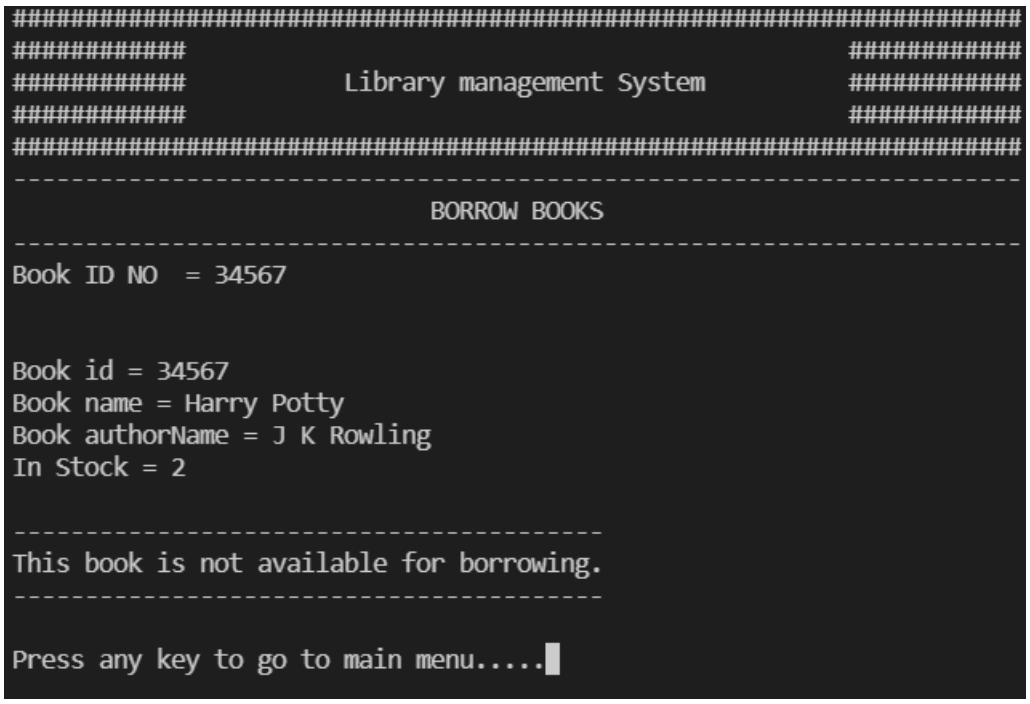
Any faculty member/ student can borrow a book from this option. There is an upper limit to the number of books that can be issued. On chosing option 3, the user will be urged to type in the ID no. of the desired book. If the given ID is present in the database, then the program will check the availability of the book. If not, then it will display “Book not found” and will be asked to try again.

If a sufficient number of copies are available, then the book name along with its other details will be displayed on the screen. If it is the desired

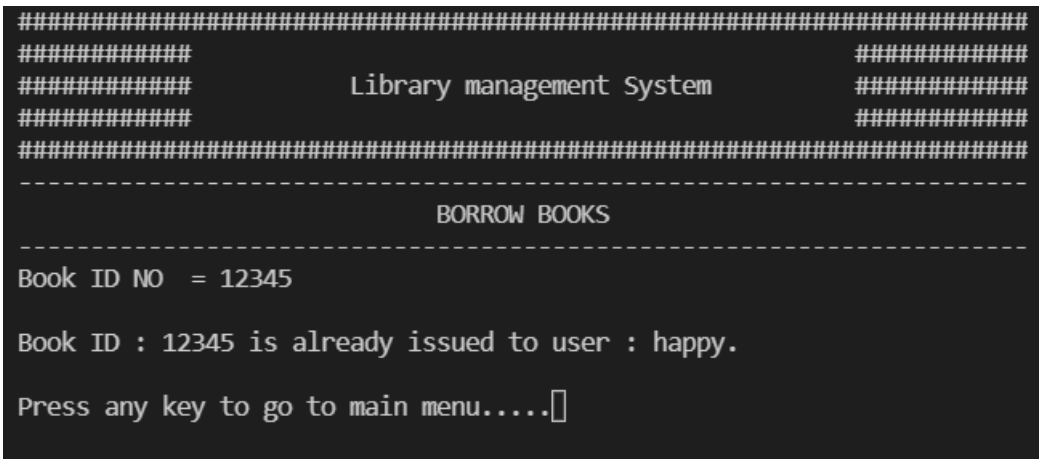
book, then the user can confirm it and the book will be issued successfully, noting the return date below.



However it is to be noted that if the number of copies available for a particular book is less than 3, then the program will only allow it to be issued to a faculty member or admin. If a student tries to issue it, the screen will display the following message.

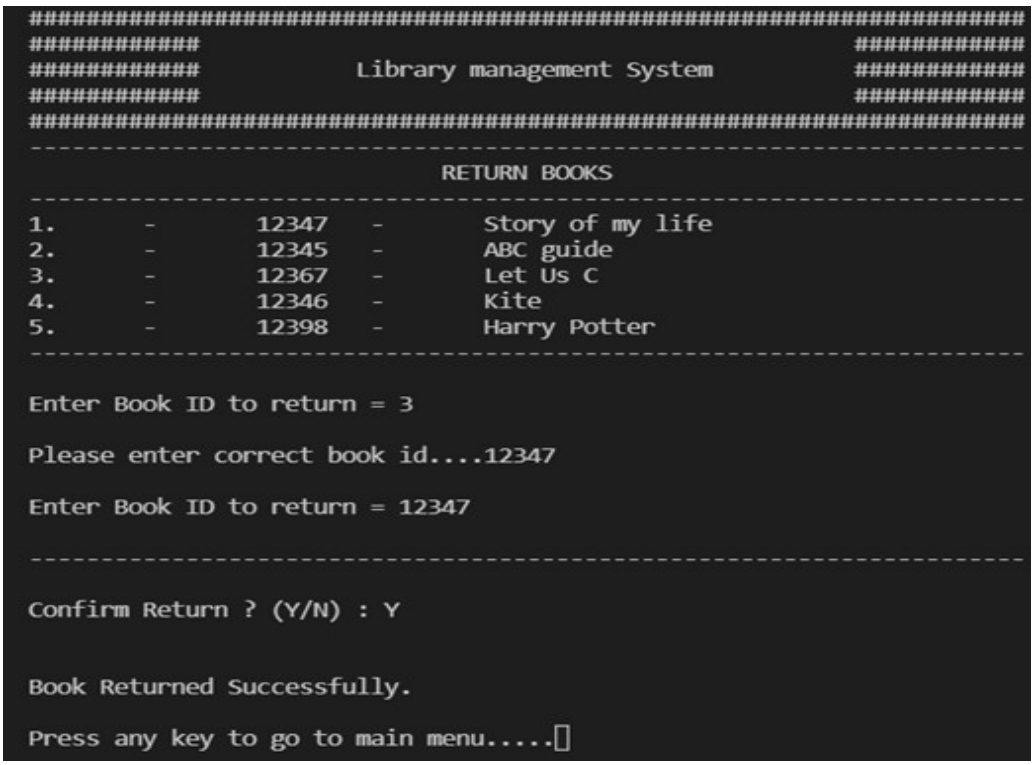


Also, the program is designed in such a way that no individual can issue the same book more than once. If a user attempts to do it, the following message will be displayed on the screen.



4.Return Book

Any book that was issued earlier can be returned through this option. On pressing 4, the terminal displays the name of the issued books along with their IDs. In order to return a particular book enter its ID and confirm the request. The following message will appear in the terminal, confirming the return.



Only for admins:

5. Add books:

Any admin can add a book into the library using this option. On pressing option 5, the admin will be urged to type in a new unique ID no. of the book that they want to add. Then the admin must give the name of the book and the name of the author. Finally the admin must specify the total number of books that they would like to add.

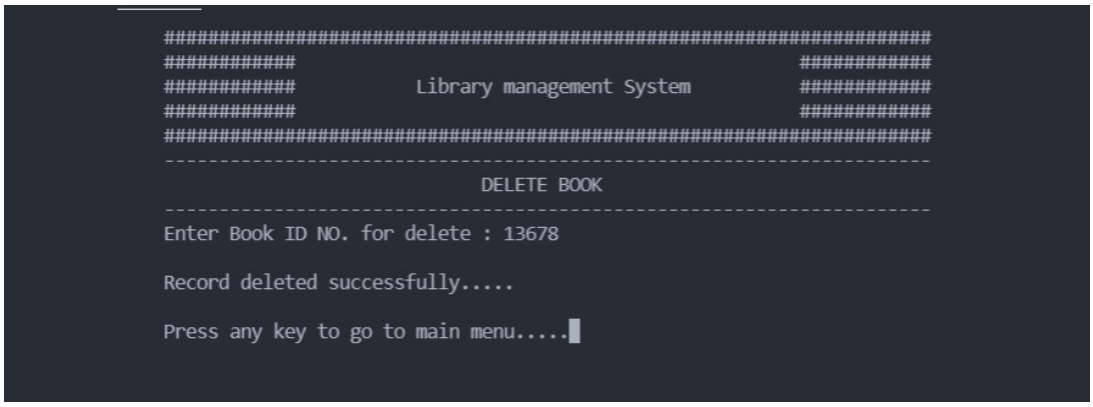
```
######################################################################
############                                              ############
############          Library management System           ############
############                                              ############
######################################################################
----------------------------------------------------------------------
                           ADD NEW BOOK
----------------------------------------------------------------------
                      ENTER BOOK DETAILS BELOW:
----------------------------------------------------------------------

Book ID NO  = 13678

Book Name  = Animal Farm

Author Name  = Orwell

Enter number of books in stock : 13
```

Back to the main menu, on pressing option 2, View Books, it can be seen that the book has been added.

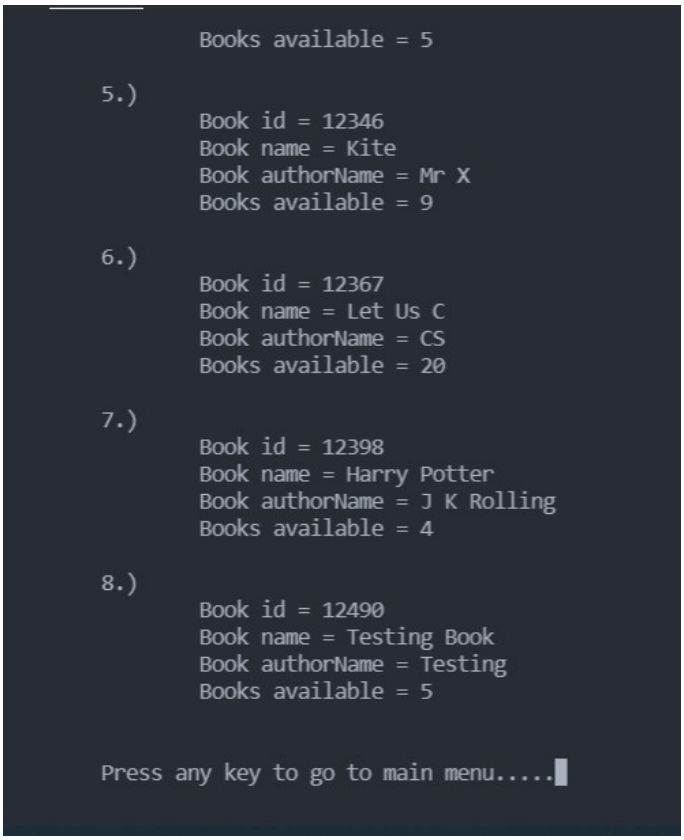
```
                Books available = 9

        6.)
                Book id = 12367
                Book name = Let Us C
                Book authorName = CS
                Books available = 20

        7.)
                Book id = 12398
                Book name = Harry Potter
                Book authorName = J K Rolling
                Books available = 4

        8.)
                Book id = 12490
                Book name = Testing Book
                Book authorName = Testing
                Books available = 5

        9.)
                Book id = 13678
                Book name = Animal Farm
                Book authorName = Orwell
                Books available = 13


        Press any key to go to main menu.....
```

6. Delete books:

The admin can also delete\remove a book from the library. On pressing option 6, the admin must specify the ID no. of the book they wish to remove.

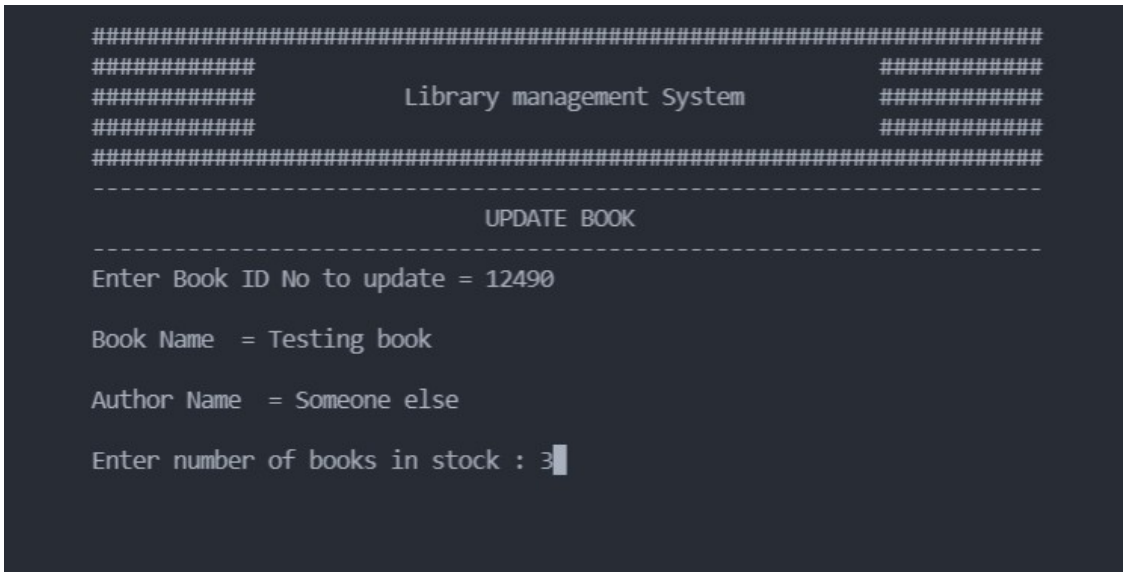


And upon viewing the books, one can see that the book has been removed.

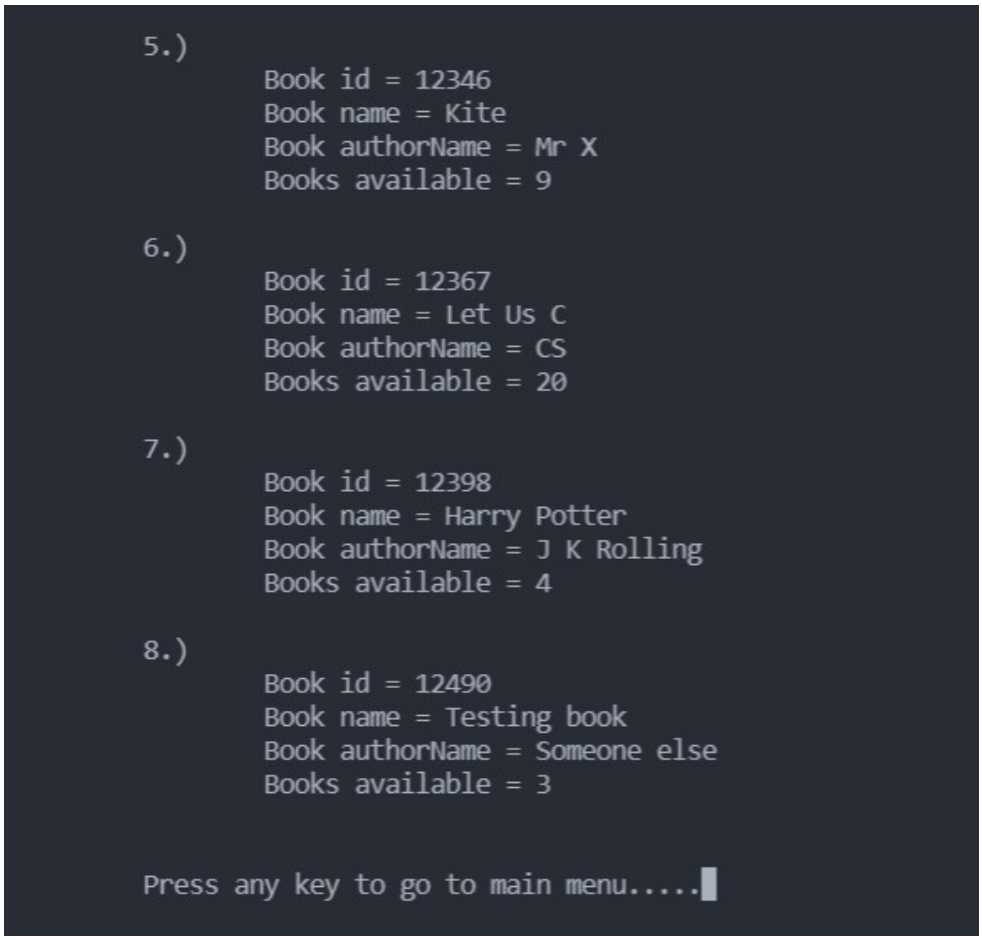


7. Update books:

The admin can also update any book they like in the library. Upon pressing option 7, the admin must specify the book ID that they wish to update. After that, they can change the name of the book, the name of the author and the available stock.



Upon viewing, we can see that it has been updated.



Once you are done with the program, press 0, Exit. “Thank You” message will be displayed in the terminal.

Pre saved login credentials: (can be manually changed by user)

admin login:
Username : admin
Password : 1234

Faculty login:
Username : teach
Password :1235

Student login:
Username : study
Password : 1236

CONTRIBUTORS

- *Shivam Kumar (19MS123) :* **Borrow, Return + combining all modules to make a single program.**
- *Swaraj Pradhan (19MS063) :* **Add, delete & update books**
- *Rishik Bhowmick (19MS064) :* **View, search**
- *Adwait Naravane Bipinchanndra (19MS151) :* **User Interface**

----------0.o----------