اونيۏرسيتي مليسيا ڤهڠ

UNIVERSITI MALAYSIA PAHANG

PUSAT SAINS MATEMATIK

MINI PROJECT BCI1023 PROGRAMMING TECHNIQUES

TITLE:

BIBLIO LIBRARY MANAGEMENT SYSTEM

LECTURER'S NAME:

DR YUSNITA BT MUHAMAD NOOR

Matric ID	Name	Section
SD22019	NUR SABIHAH BINTI ANUAR	01G
SD22035	NURUL NAJWA BINTI NORHISHAM	02G
SD22017	BATRISYIA BINTI ISMAIL	01G
SD22007	MIZA SYAZWANA BINTI MOHD SAFIAN	01G

TABLE OF CONTENTS

1.0 CASE STUDY

Biblio's Library is a modern library that aims to streamline its operations and reduce errors in managing a large collection of books and memberships. The traditional manual record-keeping system has proven to be time-consuming and prone to mistakes, especially with the increasing number of books and student memberships.

Biblio's Library has decided to develop a fully computerised Library Management System. This system will automate various tasks and provide an efficient way for the staff to manage the library. The Library Management System offers a range of functionalities to support different aspects of library operations.

The system allows staff members to add new books to the library. When adding a book, the staff enters details such as the title and publication year. This information is stored in a database, ensuring accurate and consistent records. The system also provides the ability to display book information, allowing students to access a comprehensive list of books along with their corresponding details. Students can easily browse through the available books, making it convenient for them to find the resources they need for their projects and assignments.

The Library Management System includes a search functionality that enables users to find books based on their titles or publication years. This feature saves time for both staff members and students, as they can quickly locate specific books without manually scanning through shelves or records.

To encourage responsible book handling, the system incorporates a fine calculation feature. It uses a predefined late and damage fee structure to determine fines for overdue or damaged books. When a student returns a book late or with damage, the system automatically calculates the applicable fine. This automated process eliminates manual calculations, ensuring accuracy and consistency in fine management.

The system includes a module for managing student memberships. Staff members can add students to the system and assign them library memberships. Students can then pay membership fees, which grant them access to borrowing privileges and other library services. By automating the membership management process, the system simplifies administrative tasks for staff members and provides a convenient way for students to enrol in the library.

Once students become library members, they can use their assigned credentials to access the library's resources and services. This includes borrowing books, extending loan periods, and returning books. The system keeps track of borrowed books and due dates, ensuring timely returns and preventing any potential issues or fines.

TITLE	**Harry Potter**	**Calculus**	**How To Be Rich**	**Pelukis Jalanan**
YEAR	**1997**	**2019**	**1965**	**2023**

Table 1: List of the books.

	FEE
DAMAGE	**RM 50.00**
RETURN LATE	**RM2.50**

Table 2: Fee for damage and return late for all books.

Membership = 5% x (fine payment)

Table 3: Discount for membership

2.0 OBJECTIVES

The objectives of this project are as follows:

i. To automate the process of adding new books to the library and maintaining accurate records. The system should allow staff members to easily enter book information, store it in a database, and provide functionalities to display and search for books based on title and publication year.

ii. To improve the user experience for both staff members and students. Staff members should have a user-friendly interface to manage book-related tasks efficiently, while students should be able to access book information easily and search for books of interest. By streamlining processes and providing a seamless experience, the system aims to enhance overall satisfaction and engagement.

iii. To automate the calculation of fines for late or damaged books. The system should have predefined rules and fees in place to accurately calculate fines based on the duration of delay or the extent of damage. This automation eliminates manual calculations, reduces errors, and ensures consistency in fine management.

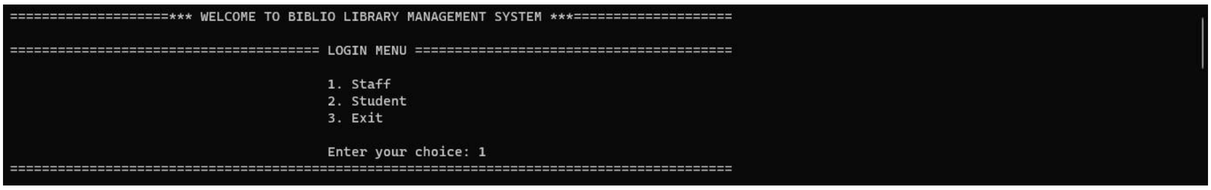
iv. To manage student memberships effectively. Staff members should be able to add students to the system, assign them library memberships, and track membership status. Students should have a streamlined process to enrol, pay membership fees, and access library services. By simplifying membership management, the system aims to improve administrative efficiency and provide a convenient experience for students.

v. To improve the efficiency and effectiveness of library operations. By automating manual tasks, reducing errors, and providing accurate and up-to-date information, the system enables staff members to spend less time on administrative work and more time on providing quality library services. This includes tasks such as book management, fine calculation, membership management, and ensuring smooth student access to library resources.

vi. To significantly reduce errors resulting from manual record-keeping. By leveraging automation and database management, the system minimises data entry errors, inconsistencies in book records, and inaccuracies in fine calculations. This objective ensures the integrity of library data and improves the reliability of library services.

3.0 EXAMPLE INPUT AND OUTPUT

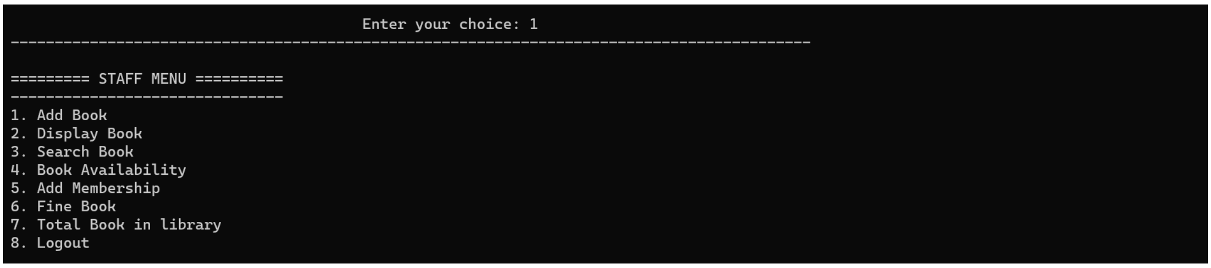
i. Greeting and Login Menu



```
====================*** WELCOME TO BIBLIO LIBRARY MANAGEMENT SYSTEM ***====================

======================================= LOGIN MENU ========================================

                                        1. Staff
                                        2. Student
                                        3. Exit

                                        Enter your choice: 1
===========================================================================================
```

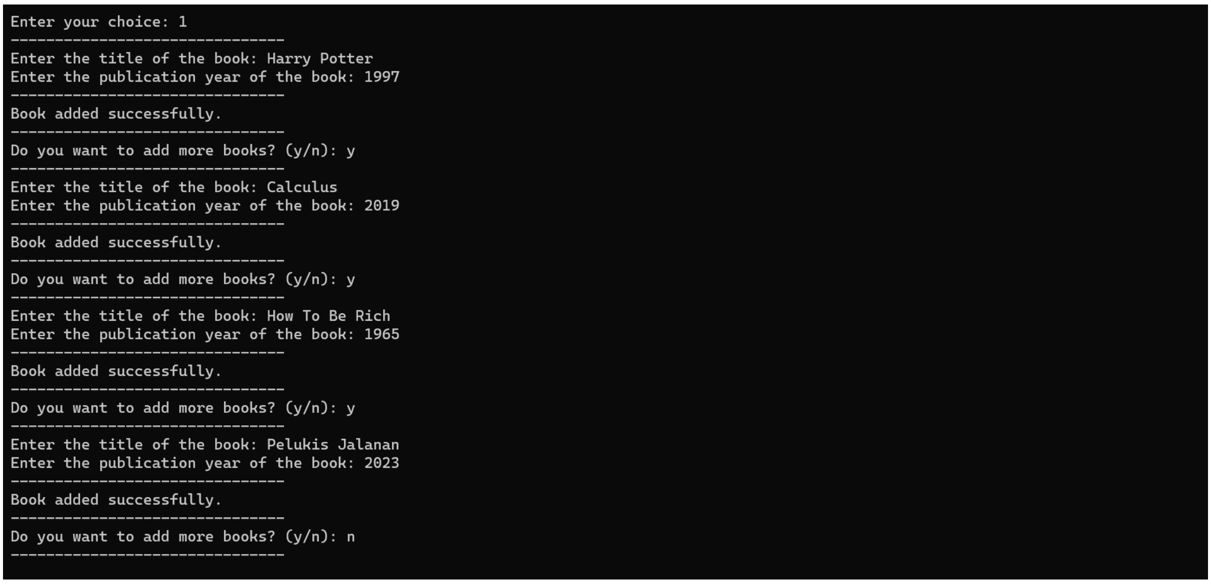
Welcome to Biblio Library Management System.

ii. Staff Menu



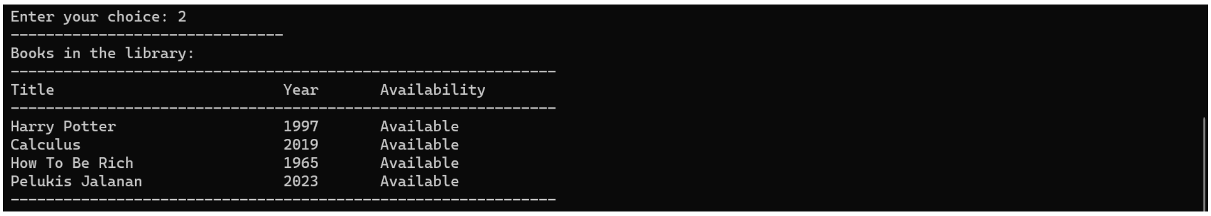
```
                                             Enter your choice: 1
-----------------------------------------------------------------------------------------------------


========= STAFF MENU ==========
-------------------------------
1. Add Book
2. Display Book
3. Search Book
4. Book Availability
5. Add Membership
6. Fine Book
7. Total Book in library
8. Logout
```

We choose 1(Staff). Users need to choose either 1-8 in the staff menu.



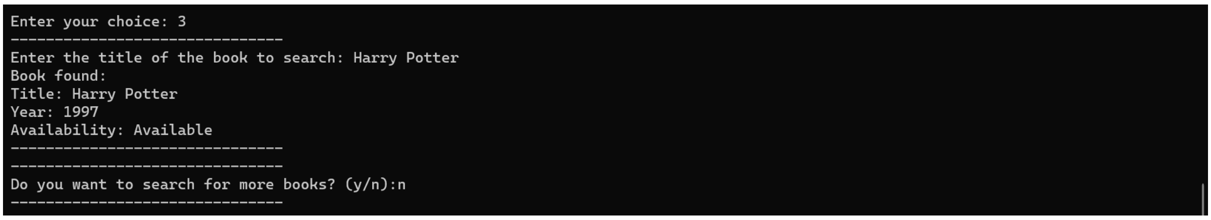
```
Enter your choice: 1
-------------------------------
Enter the title of the book: Harry Potter
Enter the publication year of the book: 1997
-------------------------------
Book added successfully.
-------------------------------
Do you want to add more books? (y/n): y
-------------------------------
Enter the title of the book: Calculus
Enter the publication year of the book: 2019
-------------------------------
Book added successfully.
-------------------------------
Do you want to add more books? (y/n): y
-------------------------------
Enter the title of the book: How To Be Rich
Enter the publication year of the book: 1965
-------------------------------
Book added successfully.
-------------------------------
Do you want to add more books? (y/n): y
-------------------------------
Enter the title of the book: Pelukis Jalanan
Enter the publication year of the book: 2023
-------------------------------
Book added successfully.
-------------------------------
Do you want to add more books? (y/n): n
-------------------------------
```

In this example, we choose 1(Add book). Users input the title of the book and the publication of the book(year).



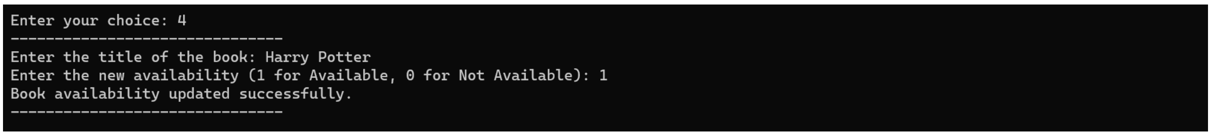
```
Enter your choice: 2
-------------------------------
Books in the library:
--------------------------------------------------------------
Title                          Year       Availability
--------------------------------------------------------------
Harry Potter                   1997       Available
Calculus                       2019       Available
How To Be Rich                 1965       Available
Pelukis Jalanan                2023       Available
--------------------------------------------------------------
```

We choose 2 (Display book). Output shows that all the books that are added are displayed.



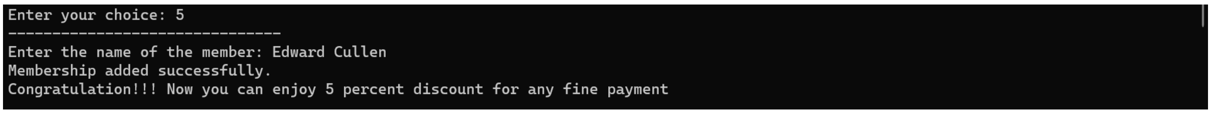
```
Enter your choice: 3
-------------------------------
Enter the title of the book to search: Harry Potter
Book found:
Title: Harry Potter
Year: 1997
Availability: Available
-------------------------------
-------------------------------
Do you want to search for more books? (y/n):n
-------------------------------
```

We choose 3(Search book). Users need to input the title of the book. Output shows the book is found and the details of the book.



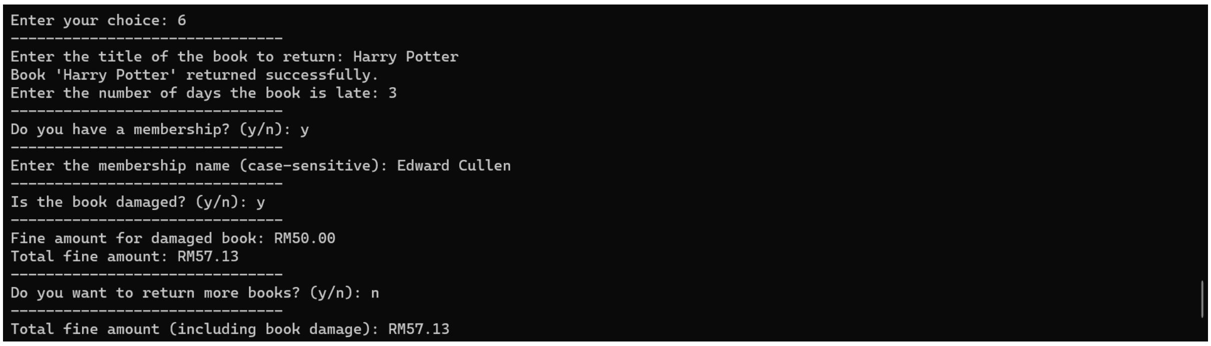
```
Enter your choice: 4
-------------------------------
Enter the title of the book: Harry Potter
Enter the new availability (1 for Available, 0 for Not Available): 1
Book availability updated successfully.
-------------------------------
```

We choose 4(Book availability). Users need to input the title of the book. Users need to input the new availability either(1 for Available, 0 for Not Available). Output shows the book is available updated successfully.



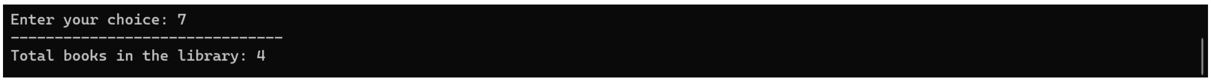
```
Enter your choice: 5
-------------------------------
Enter the name of the member: Edward Cullen
Membership added successfully.
Congratulation!!! Now you can enjoy 5 percent discount for any fine payment
```

We choose 5(Add membership). Users need to input the name of the member. Output shows membership added successfully.



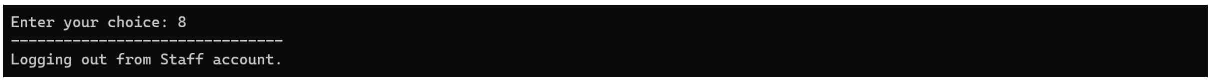
```
Enter your choice: 6
-------------------------------
Enter the title of the book to return: Harry Potter
Book 'Harry Potter' returned successfully.
Enter the number of days the book is late: 3
-------------------------------
Do you have a membership? (y/n): y
-------------------------------
Enter the membership name (case-sensitive): Edward Cullen
-------------------------------
Is the book damaged? (y/n): y
-------------------------------
Fine amount for damaged book: RM50.00
Total fine amount: RM57.13
-------------------------------
Do you want to return more books? (y/n): n
-------------------------------
Total fine amount (including book damage): RM57.13
```

We choose 6(Fine book). User need to input the title of the book return, number of days the book is late, and the membership name. User need to answer (y=yes / n=no) for membership, damaged books, and returned more books. Output shows the fine amount for damaged books, total fine amount and total fine amount (including book damage).



```
Enter your choice: 7
-------------------------------
Total books in the library: 4
```

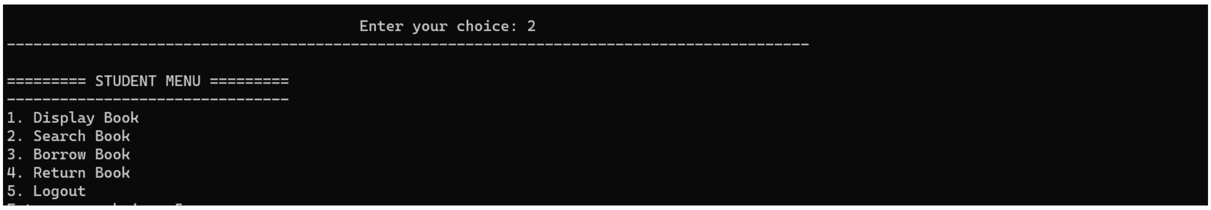
We choose 7(Total Book In Library). Output shows the total book in the library.



```
Enter your choice: 8
-------------------------------
Logging out from Staff account.
```

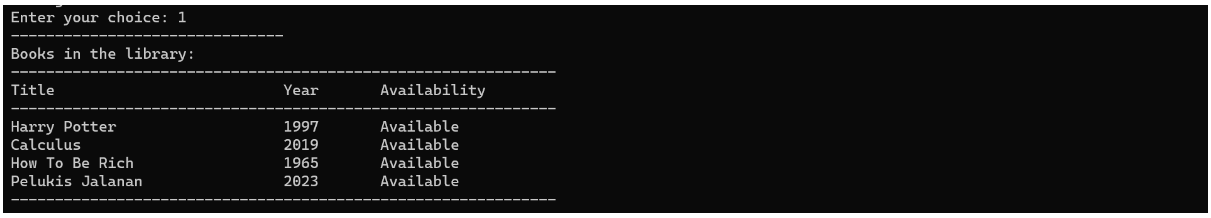
We choose 8(Log out staff).

ii) Student Menu



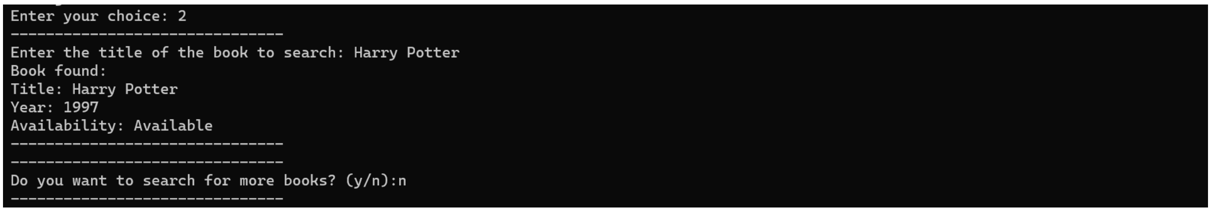
```
                                        Enter your choice: 2
-------------------------------------------------------------------------------------------

========= STUDENT MENU =========
--------------------------------
1. Display Book
2. Search Book
3. Borrow Book
4. Return Book
5. Logout
```

We choose 1(Student). User need to choose either 1-5 in the student menu.



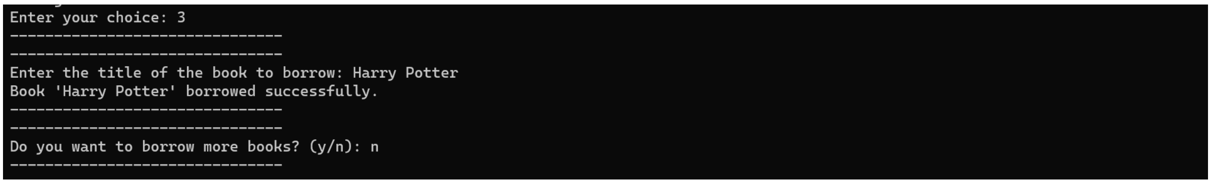
```
Enter your choice: 1
-------------------------------
Books in the library:
--------------------------------------------------------------
Title                          Year       Availability
--------------------------------------------------------------
Harry Potter                   1997       Available
Calculus                       2019       Available
How To Be Rich                 1965       Available
Pelukis Jalanan                2023       Available
--------------------------------------------------------------
```

We choose 1(Display book).Output shows that all the book that is added is displayed.



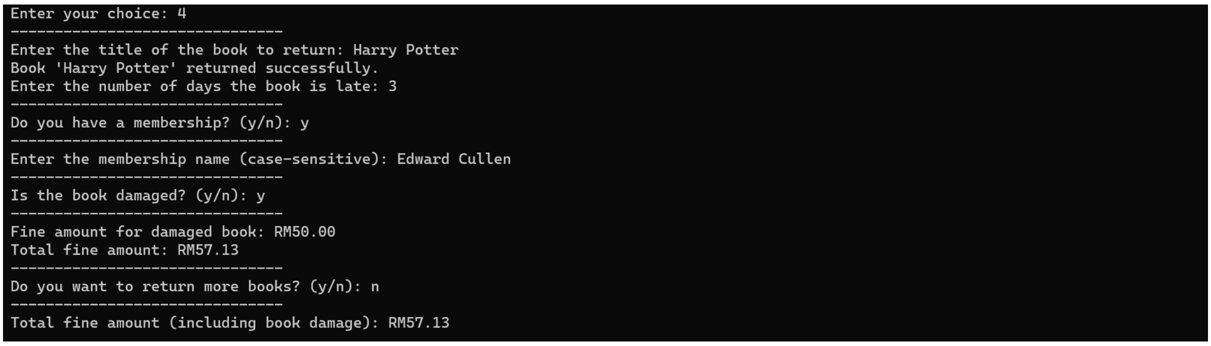
```
Enter your choice: 2
-------------------------------
Enter the title of the book to search: Harry Potter
Book found:
Title: Harry Potter
Year: 1997
Availability: Available
-------------------------------
-------------------------------
Do you want to search for more books? (y/n):n
-------------------------------
```

We choose 2(Search book). User need to input the title of the book to search. Output shows the book is found and the details of the book.



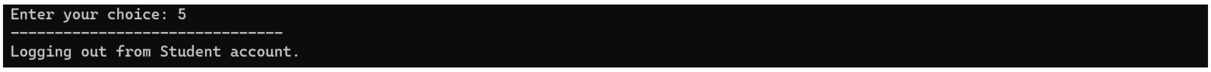
```
Enter your choice: 3
-------------------------------
-------------------------------
Enter the title of the book to borrow: Harry Potter
Book 'Harry Potter' borrowed successfully.
-------------------------------
-------------------------------
Do you want to borrow more books? (y/n): n
-------------------------------
```

We choose 3(Borrow book). User need to input the title of the book to borrow. Output shows that the book is borrowed successfully.



```
Enter your choice: 4
-------------------------------
Enter the title of the book to return: Harry Potter
Book 'Harry Potter' returned successfully.
Enter the number of days the book is late: 3
-------------------------------
Do you have a membership? (y/n): y
-------------------------------
Enter the membership name (case-sensitive): Edward Cullen
-------------------------------
Is the book damaged? (y/n): y
-------------------------------
Fine amount for damaged book: RM50.00
Total fine amount: RM57.13
-------------------------------
Do you want to return more books? (y/n): n
-------------------------------
Total fine amount (including book damage): RM57.13
```

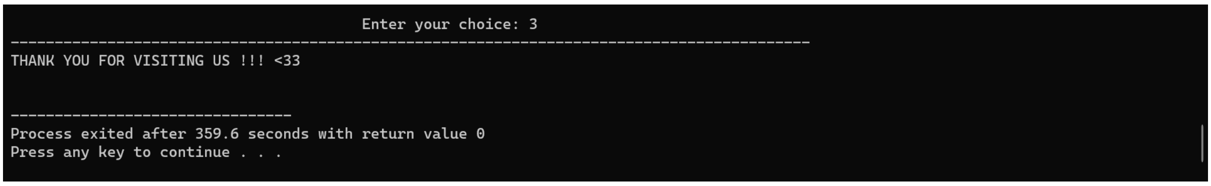
We choose 4(Return book). User need to input the title of the book return, number of days the book is late, and the membership name. User need to answer (y=yes / n=no) for membership, damaged book, returned more book. Output shows the fine amount for damaged book, total fine amount and total fine amount (including book damage).



```
Enter your choice: 5
-------------------------------
Logging out from Student account.
```

We choose 5(Log out student).

iii) Exit from the library system

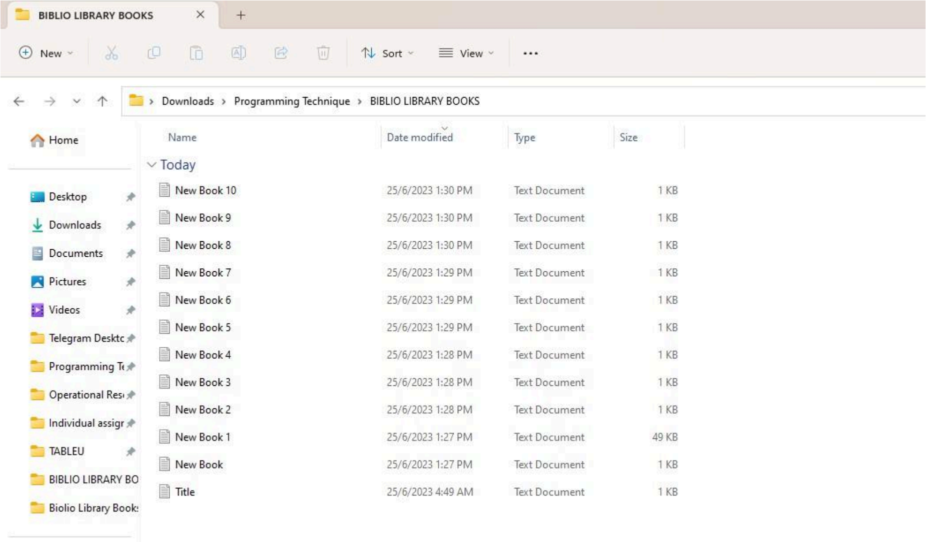


```
                                        Enter your choice: 3
------------------------------------------------------------------------------------------
THANK YOU FOR VISITING US !!! <33


--------------------------------
Process exited after 359.6 seconds with return value 0
Press any key to continue . . .
```

We choose 3(Exit).

iv) Export new book added in the library to text document (*.txt)

It will be automatically added into the folder when the staff add a new book and exit the system.

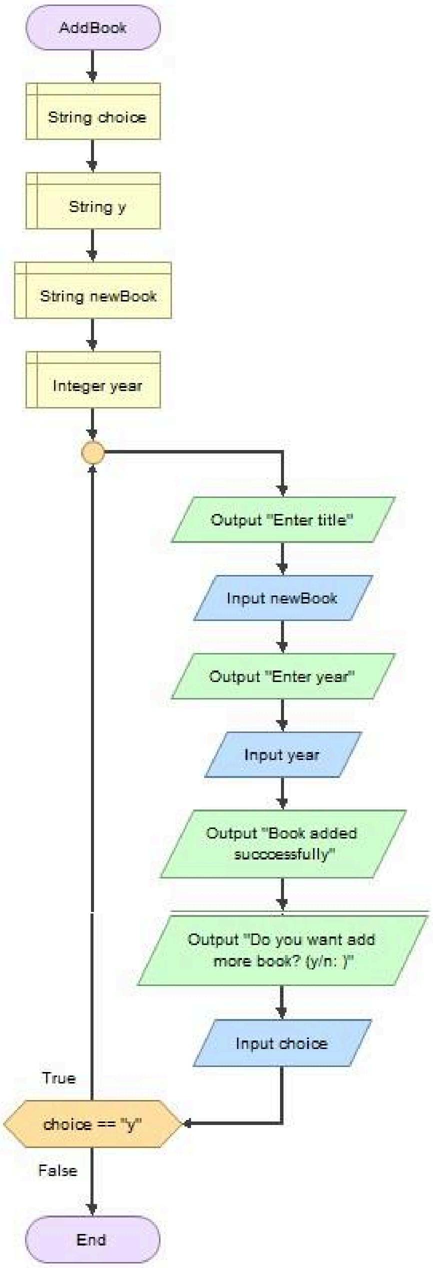


Name	Date modified	Type	Size
New Book 10	25/6/2023 1:30 PM	Text Document	1 KB
New Book 9	25/6/2023 1:30 PM	Text Document	1 KB
New Book 8	25/6/2023 1:30 PM	Text Document	1 KB
New Book 7	25/6/2023 1:29 PM	Text Document	1 KB
New Book 6	25/6/2023 1:29 PM	Text Document	1 KB
New Book 5	25/6/2023 1:29 PM	Text Document	1 KB
New Book 4	25/6/2023 1:28 PM	Text Document	1 KB
New Book 3	25/6/2023 1:28 PM	Text Document	1 KB
New Book 2	25/6/2023 1:28 PM	Text Document	1 KB
New Book 1	25/6/2023 1:27 PM	Text Document	49 KB
New Book	25/6/2023 1:27 PM	Text Document	1 KB
Title	25/6/2023 4:49 AM	Text Document	1 KB

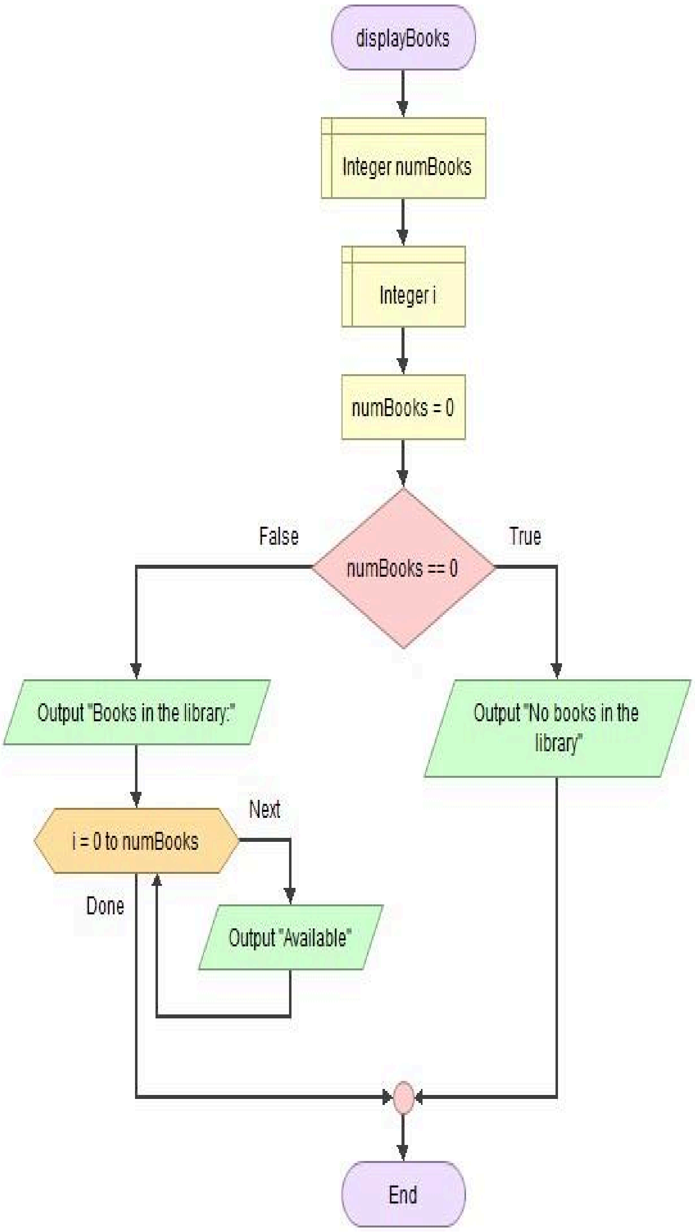
Sample of text document in Biblio Library Books folder

4.0 FLOWCHART

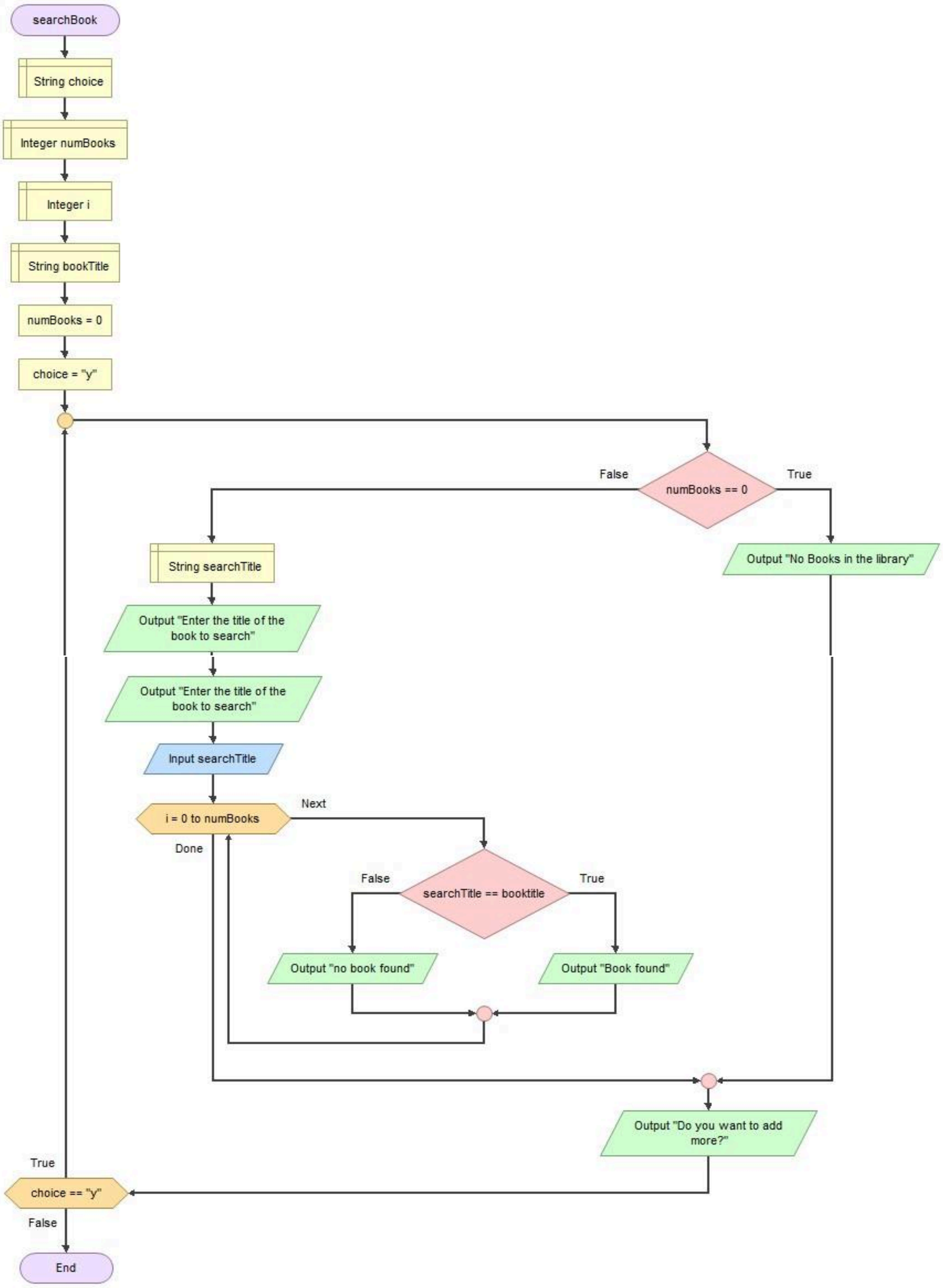
i. **void addBook()** (void with no parameter)



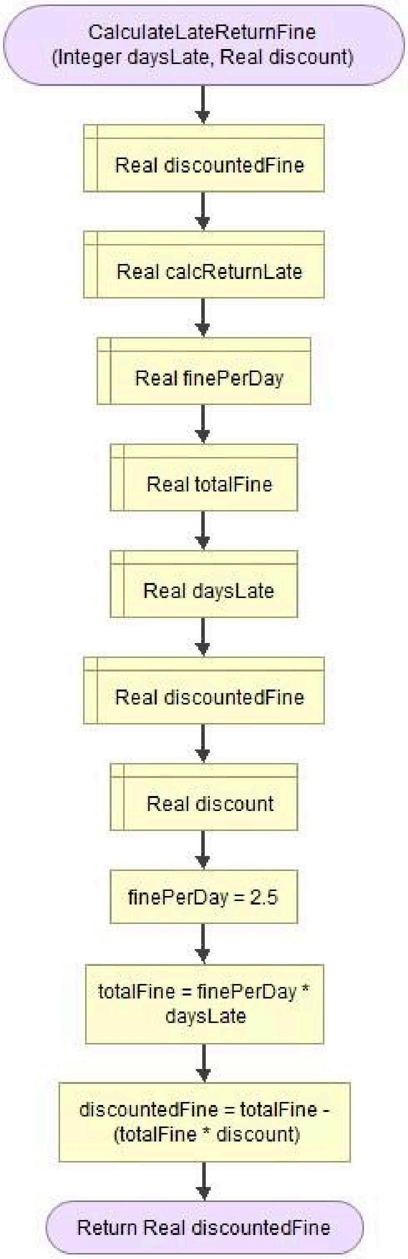
ii. void displayBooks (void with no parameter)



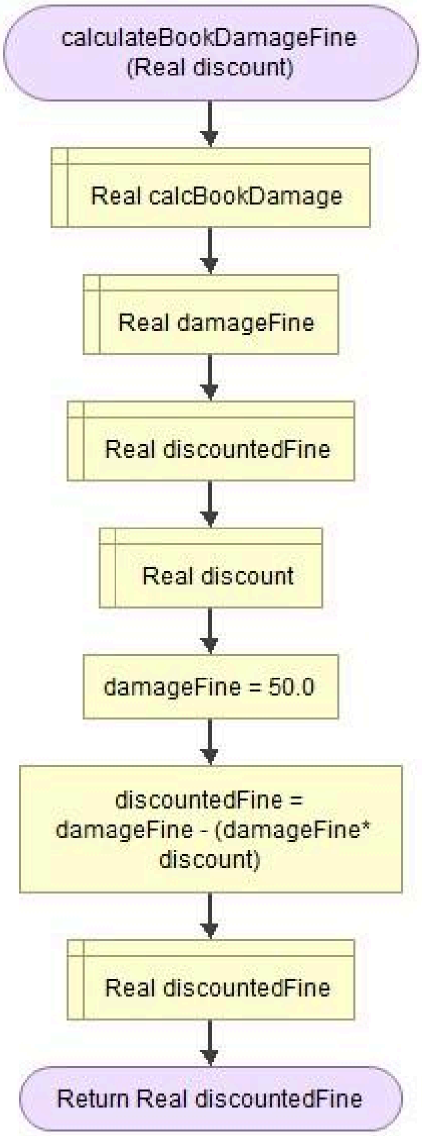
iii. void searchBooks (void with no parameter)



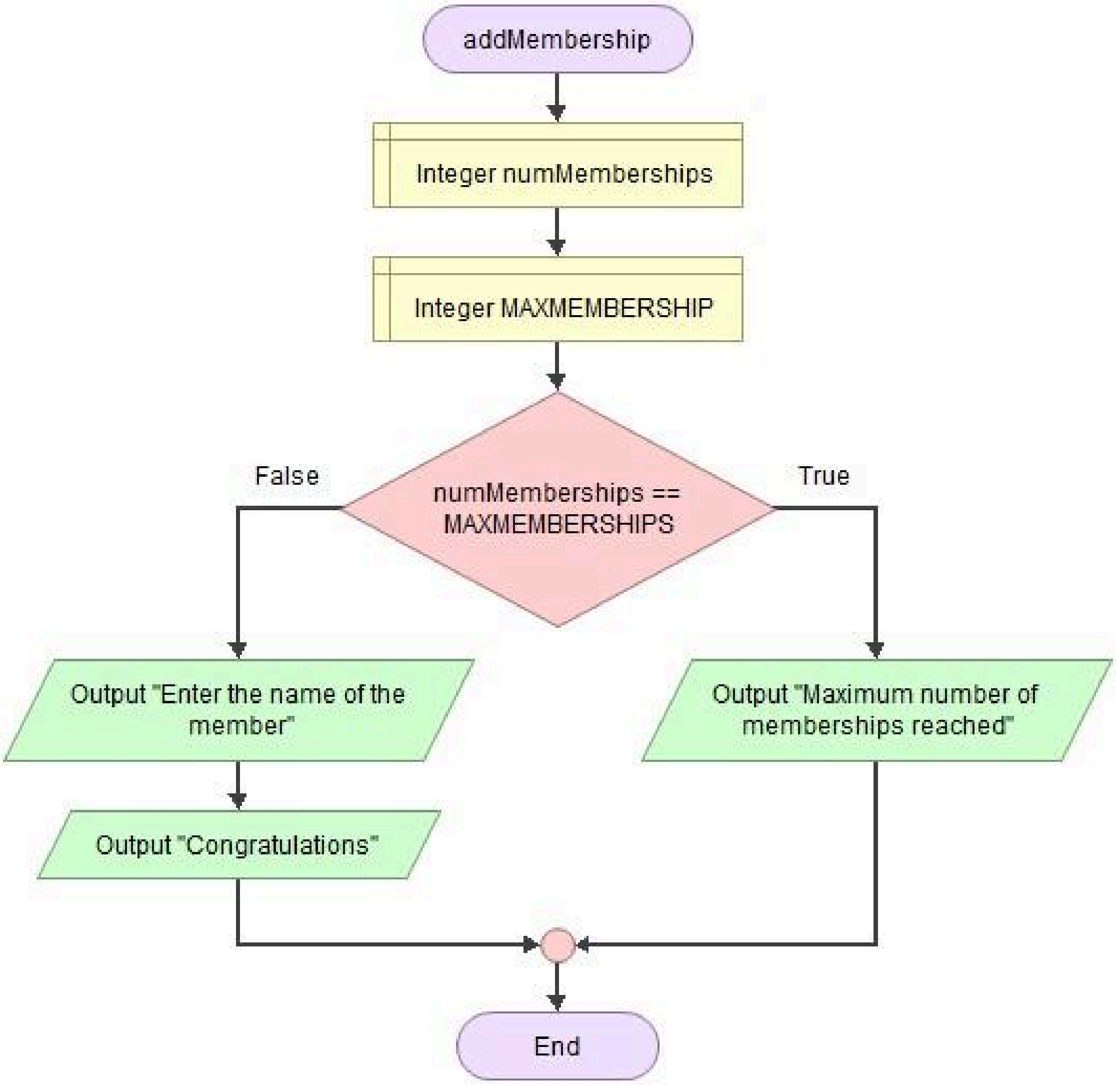
iv. float calculateLateReturnFine (int daysLate, float discount) (non-void with parameter)


CalculateLateReturnFine (Integer daysLate, Real discount)
Real discountedFine
Real calcReturnLate
Real finePerDay
Real totalFine
Real daysLate
Real discountedFine
Real discount
finePerDay = 2.5
totalFine = finePerDay * daysLate
discountedFine = totalFine - (totalFine * discount)
Return Real discountedFine

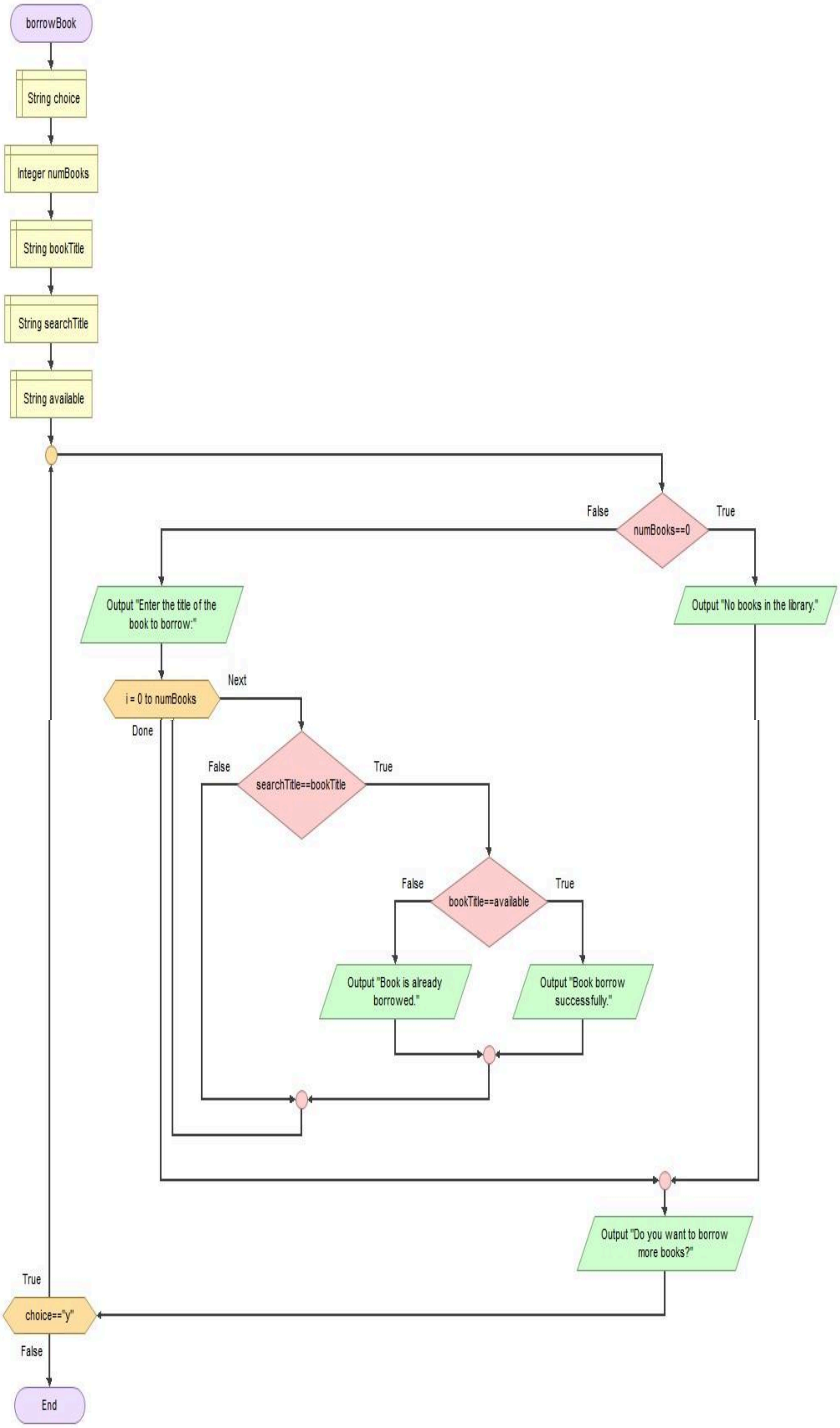
v. float calculateBookDamageFine(float discount) (non-void with parameter)



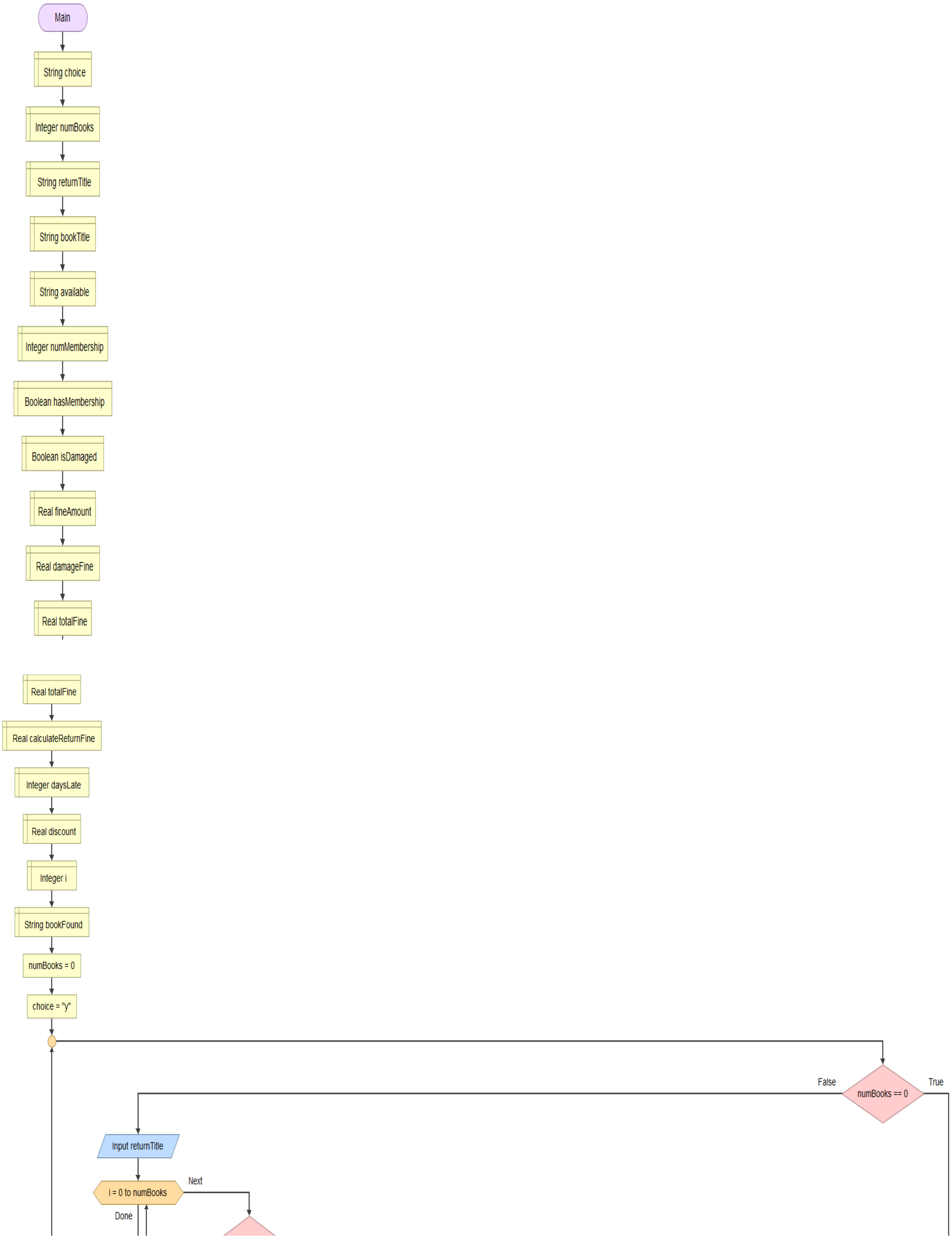
vi.void addMembership() (void with no parameter)

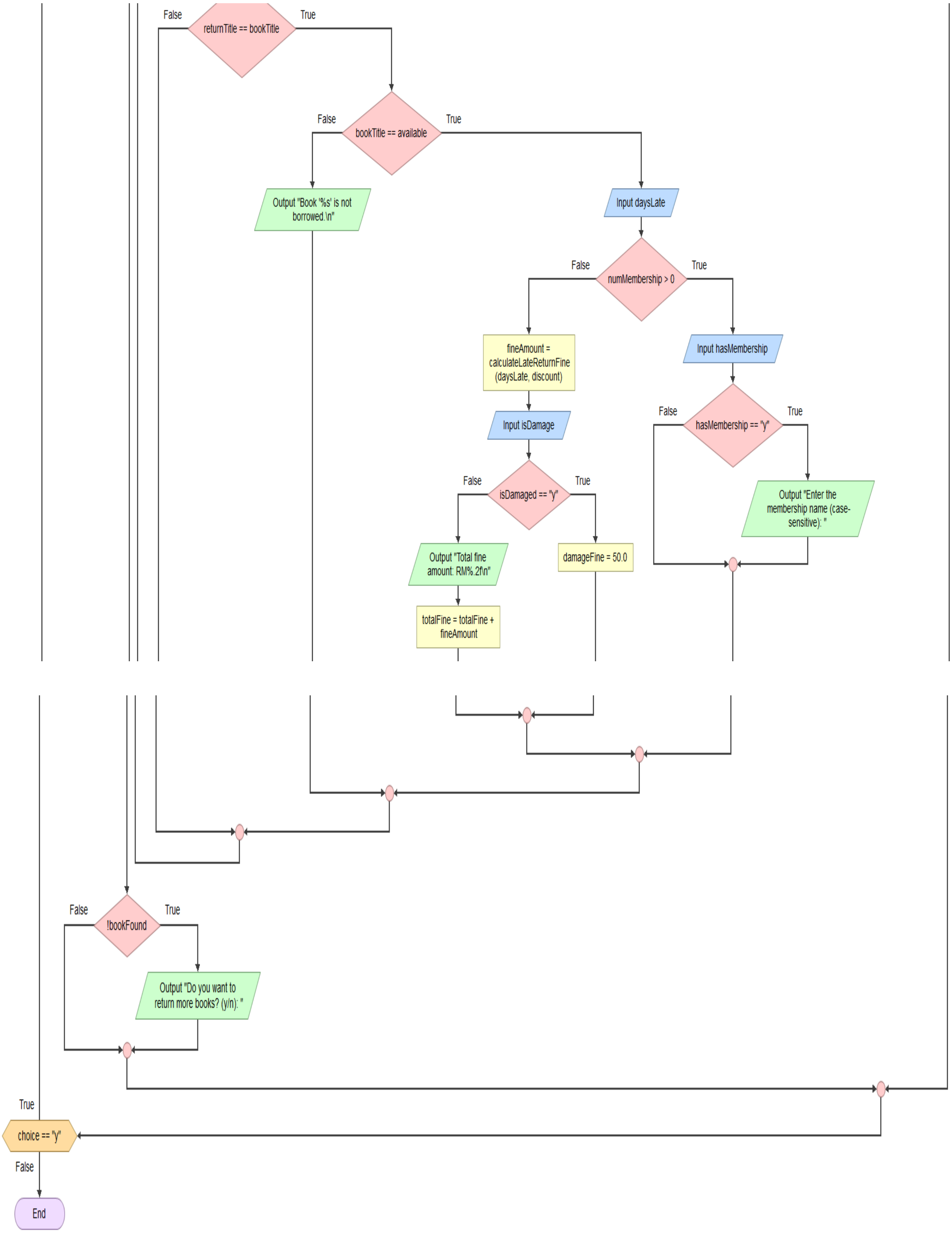


vii. void borrowBook() (void with no parameter)

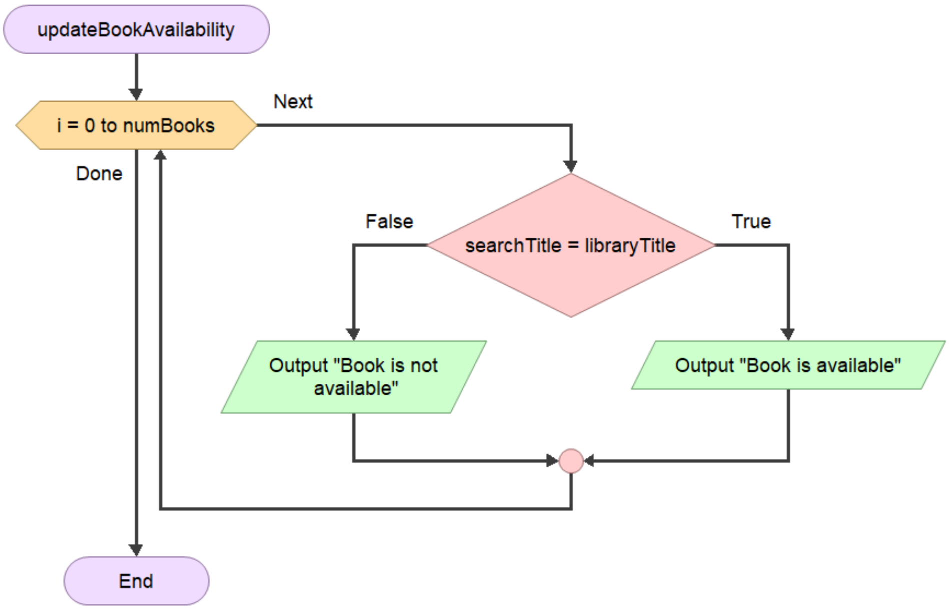


viii. void returnBook() (void with no parameter)

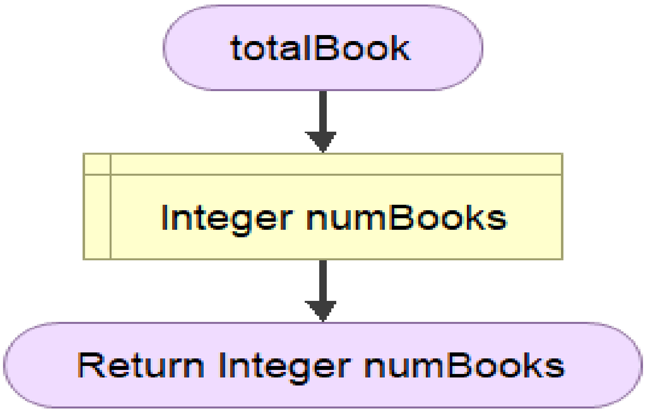



False
returnTitle == bookTitle
True
False
bookTitle == available
True
Output "Book '%s' is not borrowed.\n"
Input daysLate
False
numMembership > 0
True
fineAmount = calculateLateReturnFine (daysLate, discount)
Input hasMembership
Input isDamage
False
hasMembership == "y"
True
False
isDamaged == "y"
True
Output "Enter the membership name (case-sensitive): "
Output "Total fine amount: RM% 2f\n"
damageFine = 50.0
totalFine = totalFine + fineAmount
False
!bookFound
True
Output "Do you want to return more books? (y/n): "
True
choice == "y"
False
End

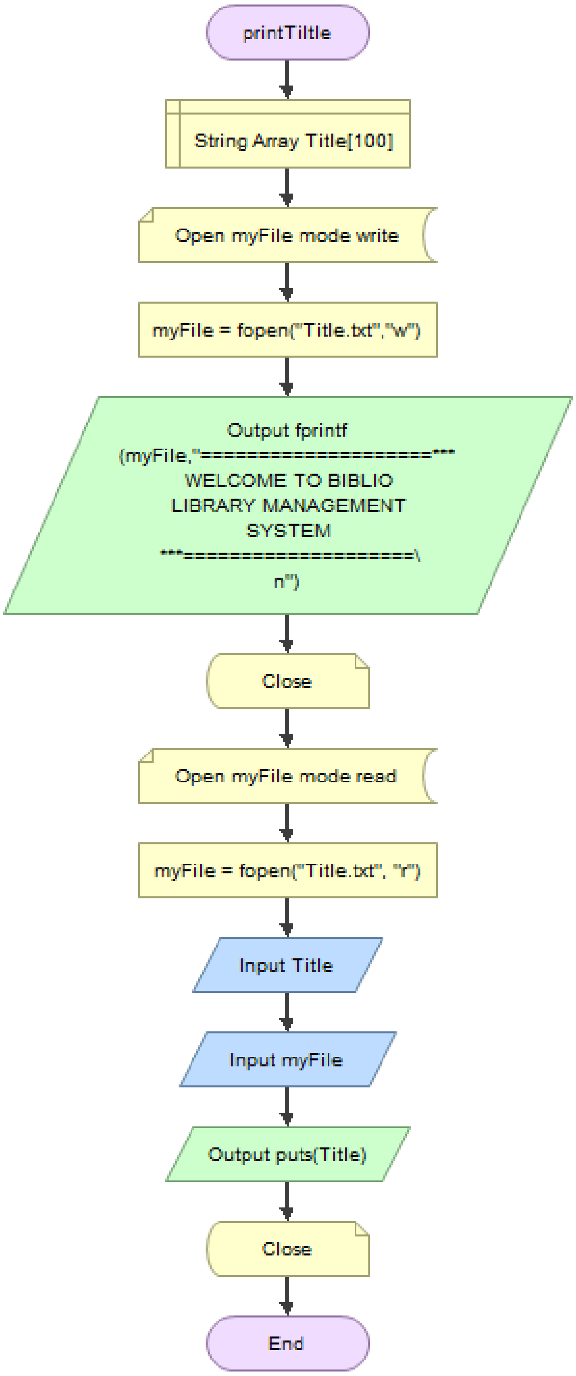
ix. void updateBookAvailability(struct Book* library, int num_Books, const char* title, int availability) (void with parameter and pointer)


updateBookAvailability
i = 0 to numBooks
Next
Done
False
searchTitle = libraryTitle
True
Output "Book is not available"
Output "Book is available"
End

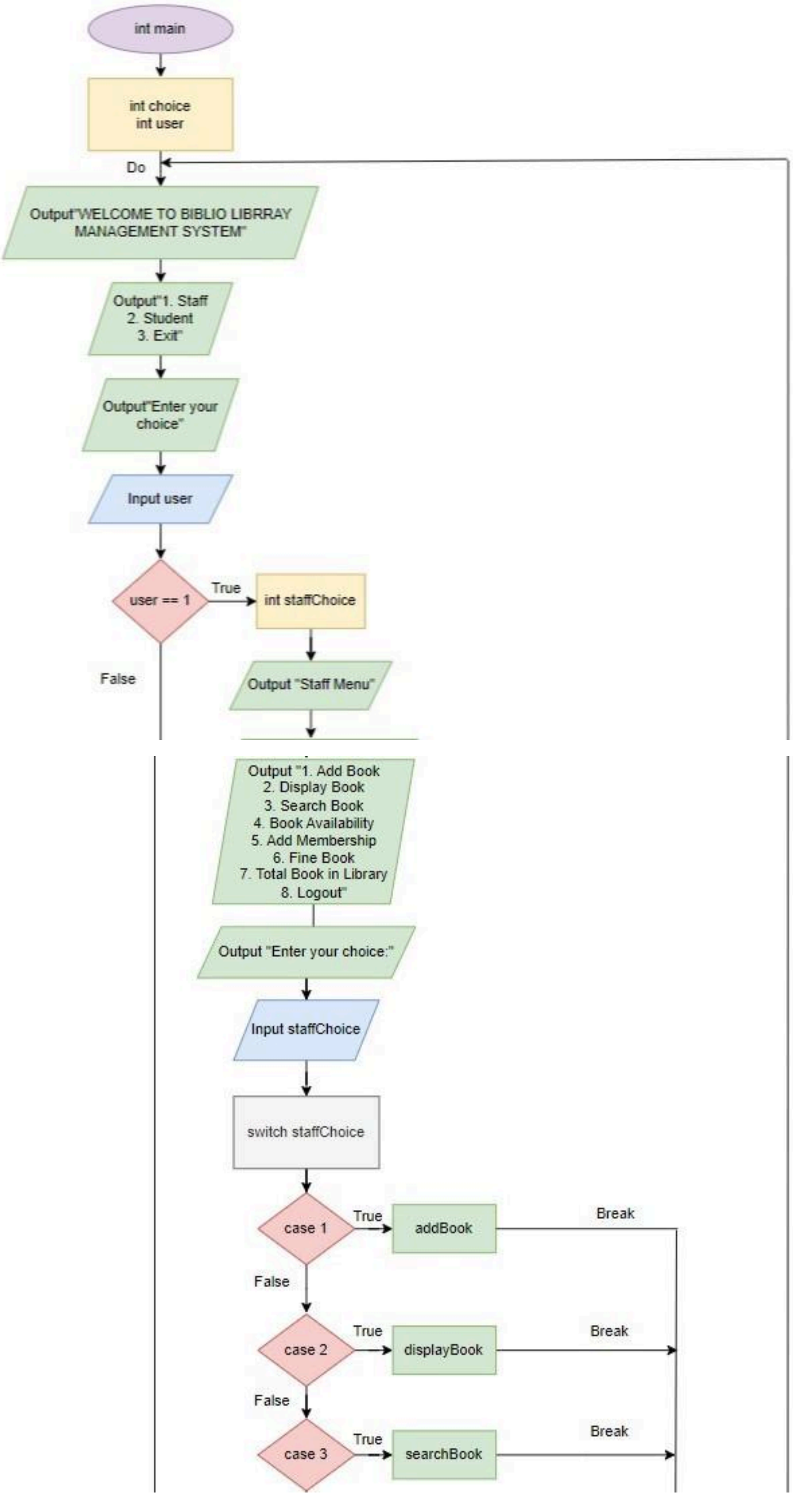
x. int totalBooks() (non-void with no parameter)

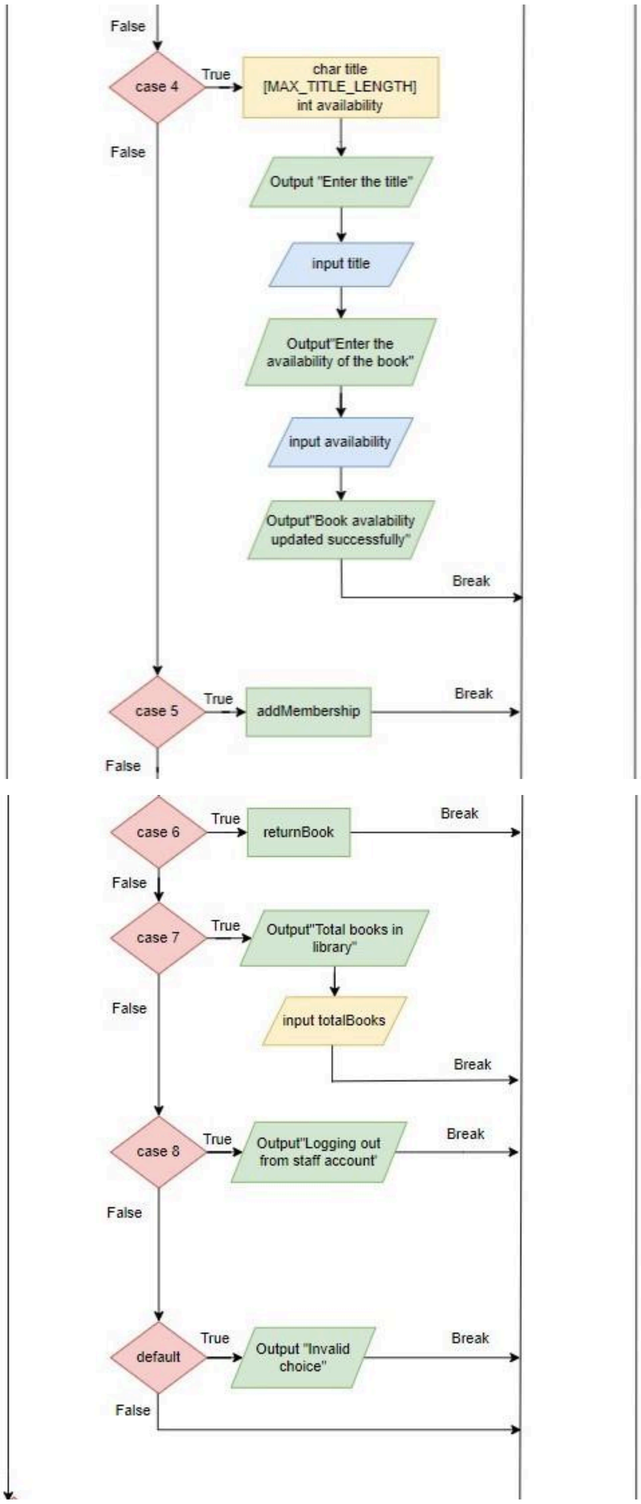


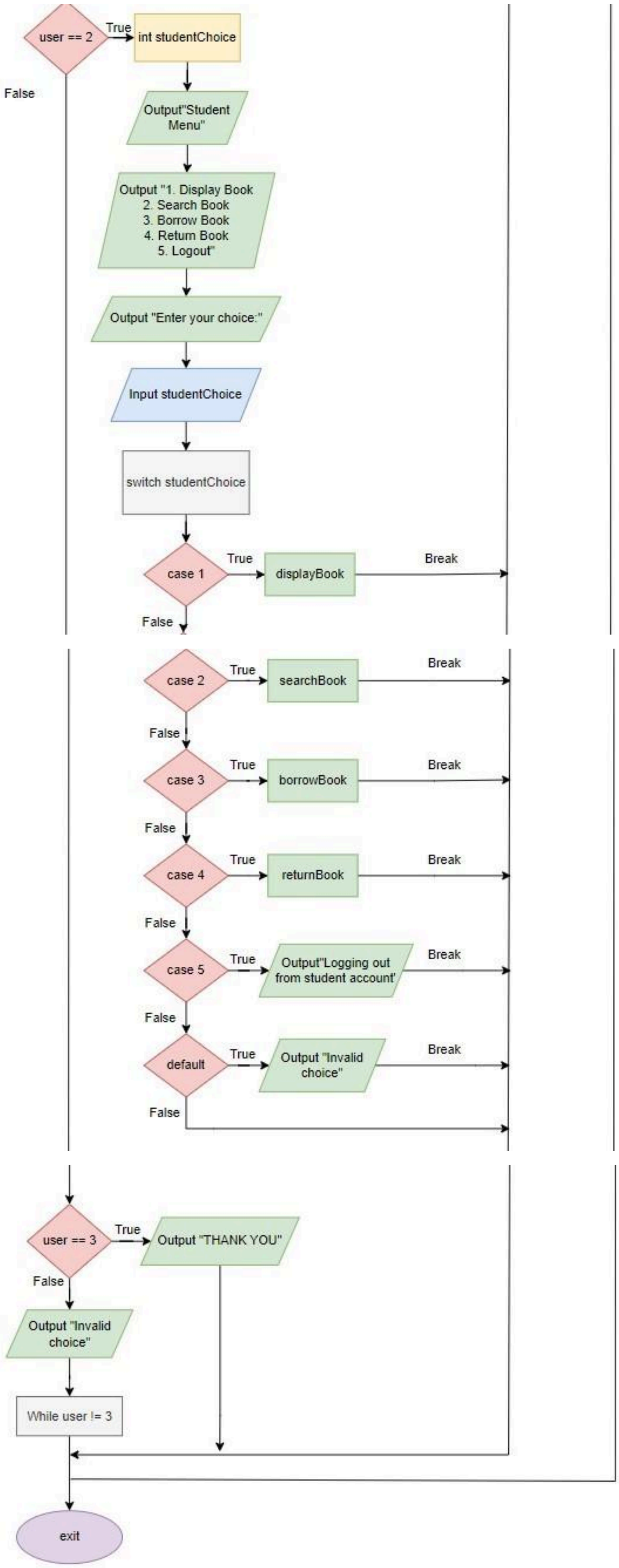
xi. void printTitle() (file operations)


printTiltle
String Array Title[100]
Open myFile mode write
myFile = fopen("Title.txt","w")
Output fprintf (myFile,"====================*** WELCOME TO BIBLIO LIBRARY MANAGEMENT SYSTEM ***====================\n")
Close
Open myFile mode read
myFile = fopen("Title.txt", "r")
Input Title
Input myFile
Output puts(Title)
Close
End

xii. int main()


int main
int choice
int user
Do
Output"WELCOME TO BIBLIO LIBRRAY MANAGEMENT SYSTEM"
Output"1. Staff
2. Student
3. Exit"
Output"Enter your choice"
Input user
user == 1
True
False
int staffChoice
Output "Staff Menu"
Output "1. Add Book
2. Display Book
3. Search Book
4. Book Availability
5. Add Membership
6. Fine Book
7. Total Book in Library
8. Logout"
Output "Enter your choice:"
Input staffChoice
switch staffChoice
case 1
True
addBook
Break
False
case 2
True
displayBook
Break
False
case 3
True
searchBook
Break


False
case 4
True
char title [MAX_TITLE_LENGTH] int availability
False
Output "Enter the title"
input title
Output"Enter the availability of the book"
input availability
Output"Book avalability updated successfully"
Break
case 5
True
addMembership
Break
False
case 6
True
returnBook
Break
False
case 7
True
Output"Total books in library"
input totalBooks
Break
False
case 8
True
Output"Logging out from staff account'
Break
False
default
True
Output "Invalid choice"
Break
False


user == 2
True
False
int studentChoice
Output"Student Menu"
Output "1. Display Book
2. Search Book
3. Borrow Book
4. Return Book
5. Logout"
Output "Enter your choice:"
Input studentChoice
switch studentChoice
case 1
True
displayBook
Break
False
case 2
True
searchBook
Break
False
case 3
True
borrowBook
Break
False
case 4
True
returnBook
Break
False
case 5
True
Output"Logging out from student account'
Break
False
default
True
Output "Invalid choice"
Break
False
user == 3
True
Output "THANK YOU"
False
Output "Invalid choice"
While user != 3
exit

5.0 CODING

```c
//biblio library system
#include <stdio.h>
#include <string.h>

#define MAX_BOOKS 100
#define MAX_TITLE_LENGTH 100
#define MAX_MEMBERSHIPS 100
#define MAX_NAME_LENGTH 100

struct Book
{
char title[MAX_TITLE_LENGTH];
int year;
int Available;
};
struct Membership
{
char name[MAX_NAME_LENGTH];
float discount;
};
struct Book library[MAX_BOOKS];
struct Membership memberships[MAX_MEMBERSHIPS];
int num_Books = 0;
int num_Memberships = 0;
//function in library system
void addBook(); //void with no parameter
void displayBooks(); //void with no parameter
void searchBook();//void with no parameter

float calculateLateReturnFine(int daysLate, float discount);//non-void with
parameter

float calculateBookDamageFine(float discount);//non-void with parameter

void addMembership();//void with no parameter

void borrowBook();//void with no parameter

void returnBook();//void with no parameter

void updateBookAvailability(struct Book* library, int num_Books, const
char* title, int availability);//void with parameter and pointer

void printTitle(); //file operations

int totalBooks(); //non-void with no parameter


// Function to add a book to the library

void addBook()

{

char choice;

do

{

struct Book new_Book;

printf("Enter the title of the book: ");

fgets(new_Book.title, sizeof(new_Book.title), stdin);

new_Book.title[strcspn(new_Book.title, "\n")] = '\0';

printf("Enter the publication year of the book: ");

scanf("%d", &new_Book.year);

getchar();

printf("-------------------------------\n");

new_Book.Available = 1;

library[num_Books++] = new_Book;

printf("Book added successfully.\n");
printf("-------------------------------\n");
printf("Do you want to add more books? (y/n): ");
scanf("%c", &choice);
getchar();
printf("-------------------------------\n");
}
// Open a new book file in write mode for each book added
FILE *Newbook = fopen(new_Book.title, "w");
if (Newbook == NULL) {
printf("Error creating file for the book.\n");
return;
}
// Write book details to the file
fprintf(Newbook, "Title: %s\nYear: %d\nAvailability: %s\n", new_Book.title,
new_Book.year, new_Book.Available ? "Available" : "Not Available");
fclose(Newbook); // Close the file
while (choice == 'y' || choice == 'Y');
}
// Function to display the list of books in the library
void displayBooks()
{
if (num_Books == 0)
{
printf("No books in the library.\n");
return;
}
printf("Books in the library:\n");
printf("===============================================================\n")
;
```

```
printf("%-30s %-10s %-12s\n", "Title", "Year", "Availability");


printf("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n")
;

for (int i = 0; i < num_Books; i++)

{

printf("%-30s %-10d %-12s\n", library[i].title, library[i].year,
library[i].Available ? "Available" : "Not Available");

}

printf("============================================================\n")
;

}

// Function to search for a book by its title

void searchBook(struct Book* library, int num_Books)

{

char choice;

do

{

if (num_Books == 0)

{

printf("No books in the library.\n");

return;

}

char searchTitle[MAX_TITLE_LENGTH];

printf("Enter the title of the book to search: ");

fgets(searchTitle, sizeof(searchTitle), stdin);

searchTitle[strcspn(searchTitle, "\n")] = '\0';

for (int i = 0; i < num_Books; i++)

{

if (strcmp(searchTitle, library[i].title) == 0)
```

```
{
printf("Book found:\n");
printf("Title: %s\n", library[i].title);
printf("Year: %d\n", library[i].year);
printf("Availability: %s\n", library[i].Available ? "Available" : "Not
Available");
printf("-------------------------------\n");
break;
}
}
printf("-------------------------------\n");
printf("Do you want to search for more books? (y/n):");
scanf("%c", &choice);
getchar();
printf("-------------------------------\n");
}while (choice == 'y' || choice == 'Y');
}
// Function to calculate fine for late return
float calculateLateReturnFine(int daysLate, float discount)
{
float finePerDay = 2.5;
float totalFine = finePerDay * daysLate;
float discountedFine = totalFine - (totalFine * discount);
return discountedFine;
}
// Function to calculate fine for book damage
float calculateBookDamageFine(float discount)
{
```

```
float damageFine = 50.0;

float discountedFine = damageFine - (damageFine * discount);

return discountedFine;

}

// Function to add a new membership

void addMembership()

{

if (num_Memberships == MAX_MEMBERSHIPS)

{

printf("Maximum number of memberships reached.\n");

return;

}

struct Membership new_Membership;

printf("Enter the name of the member: ");

fgets(new_Membership.name, sizeof(new_Membership.name), stdin);

new_Membership.name[strcspn(new_Membership.name, "\n")] = '\0';

memberships[num_Memberships++] = new_Membership;

printf("Membership added successfully.\n");

printf("Congratulation!!! Now you can enjoy 5 percent discount for any fine
payment\n");

}

// Function to borrow a book

void borrowBook()

{

char choice;

do

{

if (num_Books == 0)
```

```
{

printf("No books in the library.\n");

return;

}

char searchTitle[MAX_TITLE_LENGTH];

printf("Enter the title of the book to borrow: ");

fgets(searchTitle, sizeof(searchTitle), stdin);

searchTitle[strcspn(searchTitle, "\n")] = '\0';

for (int i = 0; i < num_Books; i++)

{

if (strcmp(searchTitle, library[i].title) == 0)

{

if (library[i].Available)

{

library[i].Available = 0;

printf("Book '%s' borrowed successfully.\n", library[i].title);

}

else

{

printf("Book '%s' is already borrowed.\n", library[i].title);

}

break;

}

}

printf("-------------------------------\n");

printf("Do you want to borrow more books? (y/n): ");

scanf("%c", &choice);

getchar();
```

```
printf("-------------------------------\n");
}
while (choice == 'y' || choice == 'Y');
}
// Function to return a book
void returnBook()
{
char choice;
float totalFine = 0.0;
do
{
if (num_Books == 0)
{
printf("No books in the library.\n");
return;
}
char returnTitle[MAX_TITLE_LENGTH];
printf("Enter the title of the book to return: ");
fgets(returnTitle, sizeof(returnTitle), stdin);
returnTitle[strcspn(returnTitle, "\n")] = '\0';
int bookFound = 0;
for (int i = 0; i < num_Books; i++)
{
if (strcmp(returnTitle, library[i].title) == 0)
{
if (!library[i].Available)
{
library[i].Available = 1;
```

```
printf("Book '%s' returned successfully.\n", library[i].title);

bookFound = 1;

// Calculate fine for late return

int daysLate;

printf("Enter the number of days the book is late: ");

scanf("%d", &daysLate);

getchar();

float discount = 0.0;

if (num_Memberships > 0)

{


printf("-------------------------------\n");

printf("Do you have a membership? (y/n): ");

char hasMembership;

scanf(" %c", &hasMembership);

getchar();

printf("-------------------------------\n");

if (hasMembership == 'y' || hasMembership == 'Y')

{

 printf("Enter the membership name (case-sensitive): ");

 char membershipName[MAX_NAME_LENGTH];

 fgets(membershipName, sizeof(membershipName), stdin);

 membershipName[strcspn(membershipName, "\n")] = '\0';

 discount = 0.05;

 }

 }

float fineAmount = calculateLateReturnFine(daysLate, discount);

char isDamaged;
```

```
printf("-------------------------------\n");
printf("Is the book damaged? (y/n): ");
scanf(" %c", &isDamaged);
getchar();
printf("-------------------------------\n");
if (isDamaged == 'y' || isDamaged == 'Y')
{
float damageFine = 50.0;
printf("Fine amount for damaged book: RM%.2f\n", damageFine);
fineAmount += damageFine;
}
printf("Total fine amount: RM%.2f\n", fineAmount);
totalFine += fineAmount;
break;
}
else
{
printf("Book '%s' is not borrowed.\n", library[i].title);
bookFound = 1;
break;
}
}
}
if (!bookFound)
{
printf("Book '%s' not found.\n", returnTitle);
}
printf("-------------------------------\n");
printf("Do you want to return more books? (y/n): ");
scanf(" %c", &choice);
getchar();
printf("-------------------------------\n");
}
while (choice == 'y' || choice == 'Y');
printf("Total fine amount (including book damage): RM%.2f\n", totalFine);
}


// Function to update a book's availability
void updateBookAvailability(struct Book* library, int num_Books, const
char* title, int availability) {
for (int i = 0; i < num_Books; i++) {
if (strcmp(title, library[i].title) == 0) {
library[i].Available = availability;
break;}
}
}
// Function to calculate the total number of books in the library
int totalBooks()
{
return num_Books;
}
// Function to declare file and print title
void printTitle()
{
FILE *myFile;
char Title[100];
```

```
myFile = fopen("Title.txt", "w"); // Open the text file

fprintf(myFile, "====================*** WELCOME TO BIBLIO LIBRARY
MANAGEMENT SYSTEM ***====================\n"); // Write to the file

fclose(myFile);

myFile = fopen("Title.txt", "r"); // Open the file for reading

fgets(Title, 100, myFile);

puts(Title); // Print the title to the top of the screen

fclose(myFile); // Close the file

}


int main()

{

int choice;

int user;

printTitle(); // Call the function to print the title

do

{

printf("======================================= LOGIN MENU
========================================\n");

printf("\t\t\t\t\t1. Staff\n");

printf("\t\t\t\t\t2. Student\n");

printf("\t\t\t\t\t3. Exit\n\n");

printf("\t\t\t\t\tEnter your choice: ");

scanf("%d", &user);

printf("===================================================================
========================\n");

getchar();

if (user == 1)

{
```

```
int staffChoice;

printf("\n========= STAFF MENU ==========\n");

printf("1. Add Book\n");

printf("2. Display Book\n");

printf("3. Search Book\n");

printf("4. Book Availability\n");

printf("5. Add Membership\n");

printf("6. Fine Book\n");

printf("7. Total Book in library\n");

printf("8. Logout\n");

printf("\nEnter your choice: ");

scanf("%d", &staffChoice);

printf("-------------------------------\n");

getchar();

switch (staffChoice)

{

case 1:

addBook();

break;

case 2:

displayBooks();

break;

case 3:

searchBook();

break;

case 4:

{

char title[MAX_TITLE_LENGTH];
int availability;
printf("Enter the title of the book: ");
fgets(title, sizeof(title), stdin);
title[strcspn(title, "\n")] = '\0';
printf("Enter the new availability (1 = Available, 0 = Not Available): ");
scanf("%d", &availability);
getchar();
updateBookAvailability(library, num_Books, title, availability);
printf("Book availability updated successfully.\n");
printf("-------------------------------\n");
}
break;
case 5:
addMembership();
break;
case 6:
returnBook();
break;
case 7:
printf("Total books in the library: %d\n", totalBooks());
break;
case 8:
printf("Logging out from Staff account.\n");
break;
default:
printf("Invalid choice. Please try again.\n");
break;
}
```

```
}

else if (user == 2)

{

int studentChoice;

printf("\n========= STUDENT MENU =========\n");

printf("1. Display Book\n");

printf("2. Search Book\n");

printf("3. Borrow Book\n");

printf("4. Return Book\n");

printf("5. Logout\n");

printf("Enter your choice: ");

scanf("%d", &studentChoice);

printf("-------------------------------\n");

getchar();

switch (studentChoice)

{

case 1:

displayBooks();

break;

case 2:

searchBook();

break;

case 3:

borrowBook();

break;

case 4:

returnBook();

break;
case 5:

printf("Logging out from Student account.\n");

break;

default:

printf("Invalid choice. Please try again.\n");

break;

}

}

else if (user == 3)

{

printf("THANK YOU FOR VISITING US !!! <33\n");

printf("Please Come Again Later XOXO\n");

}

else

{

printf("Invalid choice. Please try again.\n");

}

printf("\n");

}

while (user != 3);

return 0;

}
```

6.0 CONCLUSION

By offering a fully functional database that enables library employees to quickly check book information and streamline processes, our programme revolutionises library management. Staff can better serve students if they have quick access to information about book availability and status. The program's thorough cataloguing system makes sure that books can be found and arranged conveniently, maximising the library's collection. The programme also creates reports and analytics to aid in making wise decisions. The user-friendly interface makes it easier for students to find books, make reservations, and get notifications, which improves their borrowing experience. Overall, our programme provides a quick and easy method for managing libraries that is advantageous to both staff and students.

7.0 REFERENCES

Anurag. (2020, December 21). Library Management System | Library Management Software in India. SkoolBeep. https://www.skoolbeep.com/blog/library-management-system/

Wikipedia Contributors. (2019, September 29). Integrated library system. Wikipedia; Wikimedia Foundation.

https://en.wikipedia.org/wiki/Integrated_library_system

MODULE -5A. (n.d.).

https://nios.ac.in/media/documents/SrSecLibrary/LCh-015A.pdf

Library Management System - an overview | ScienceDirect Topics. (n.d.). Www.sciencedirect.com.

Library Management System - an overview | ScienceDirect Topics

School Library Management System – Library Management Software. (n.d.). Www.elibrarysoftware.com. Retrieved June 24, 2023, from

School Library Management System