// PROJECT

// LIBRARY MANAGEMENT SYSTEM

/\*for adding books, searching books, issuing books to members,and returning books.\*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define MAX\_BOOKS 100

#define MAX\_MEMBERS 100

// now defining structures to represent the books and members

struct Book {

int bookID;

char title[50];

char author[50];

int available;

};

struct Member {

int memberID;

char name[50];

int borrowedBooks[MAX\_BOOKS];

int numBorrowedBooks;

};

// creating storage space for books and members

struct Book libraryBooks[MAX\_BOOKS];

struct Member libraryMembers[MAX\_MEMBERS];

int numBooks = 0;

int numMembers = 0;

// NOW ADDING BOOK

void addBook() {

struct Book newBook;

printf("Enter Book ID: ");

scanf("%d", &newBook.bookID);

printf("Enter Book Title: ");

scanf(" %[^\n]", newBook.title);

printf("Enter Book Author: ");

scanf(" %[^\n]", newBook.author);

newBook.available = 1;

libraryBooks[numBooks++] = newBook;

printf("Book added successfully!\n");

}

// FOR SEARCHING

void searchBook()

{

char keyword[50];

printf("Enter search keyword: ");

scanf(" %[^\n]", keyword);

int found = 0;

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

{

if (strstr(libraryBooks[i].title, keyword) || strstr(libraryBooks[i].author, keyword))

{

printf("Book ID: %d\n", libraryBooks[i].bookID);

printf("Title: %s\n", libraryBooks[i].title);

printf("Author: %s\n", libraryBooks[i].author);

printf("Available: %s\n", libraryBooks[i].available ? "Yes" : "No");

found = 1;

}

}

if (!found)

{

printf("No matching books found.\n");

}

}

// FOR ISSUING

void issueBook() {

int memberID, bookID;

printf("Enter Member ID: ");

scanf("%d", &memberID);

printf("Enter Book ID: ");

scanf("%d", &bookID);

struct Member \*member = NULL;

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

{

if (libraryMembers[i].memberID == memberID)

{

member = &libraryMembers[i];

break;

}

}

if (member == NULL)

{

printf("Member not found.\n");

return;

}

if (member->numBorrowedBooks >= MAX\_BOOKS) {

printf("Member has reached the borrowing limit.\n");

return;

}

struct Book \*book = NULL;

for (int i = 0; i < numBooks; i++) {

if (libraryBooks[i].bookID == bookID) {

book = &libraryBooks[i];

break;

}

}

if (book == NULL) {

printf("Book not found.\n");

return;

}

if (!book->available) {

printf("Book is not available for borrowing.\n");

return;

}

book->available = 0;

member->borrowedBooks[member->numBorrowedBooks++] = bookID;

printf("Book issued successfully!\n");

}

// FOR RETURNING

void returnBook() {

int memberID, bookID;

printf("Enter Member ID: ");

scanf("%d", &memberID);

printf("Enter Book ID: ");

scanf("%d", &bookID);

struct Member \*member = NULL;

for (int i = 0; i < numMembers; i++) {

if (libraryMembers[i].memberID == memberID) {

member = &libraryMembers[i];

break;

}

}

if (member == NULL) {

printf("Member not found.\n");

return;

}

struct Book \*book = NULL;

for (int i = 0; i < numBooks; i++) {

if (libraryBooks[i].bookID == bookID) {

book = &libraryBooks[i];

break;

}

}

if (book == NULL) {

printf("Book not found.\n");

return;

}

int found = 0;

for (int i = 0; i < member->numBorrowedBooks; i++) {

if (member->borrowedBooks[i] == bookID) {

found = 1;

member->numBorrowedBooks--;

member->borrowedBooks[i] = member->borrowedBooks[member->numBorrowedBooks];

break;

}

}

if (!found) {

printf("Member has not borrowed this book.\n");

return;

}

book->available = 1;

printf("Book returned successfully!\n");

}

8

int main() {

int choice;

do {

printf("\n===== Library Management System =====\n");

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

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

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

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

printf("5. Exit\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch (choice)

{

case 1:

addBook();

break;

case 2:

searchBook();

break;

case 3:

issueBook();

break;

case 4:

returnBook();

break;

case 5:

printf("Exiting...\n");

break;

default:

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

break;

}

} while (choice != 5);

return 0;

}