

BACHELOR OF SCIENCE IN

COMPUTER SCIENCE AND ENGINEERING

**PROJECT REPORT**

Course Code: CSE-218

Course Title: Data Structure Lab



**LIBRARY MANAGEMENT SYSTEM**

|  |  |  |
| --- | --- | --- |
| **SUBMITTED BY** | | **SUBMITTED TO** |
| Name | Nusrat Jahan Prity | Abdur Rahman Riad  Lab Technical Officer (LTO)  Department of CSE  Feni University |
| ID | 222031017 |
| Batch | 28th |
| Department of CSE  Feni University | |

Date of Submission: 23 August 2023

* **INTRODUCTION**

Library management is a project that manages and stores books information electronically according to students needs .The system helps both students and library manager to keep a constant track of all the books available in the library. It allows both the admin and the student to search for the desired book. The project titled Library Management System is Library Management software for monitoring and controlling the transactions in a library. The project "Library Management System" is developed in C, which mainly focuses on basic operations in a library like adding new books, and updating new information, searching books and members and return books.

* **OBJECTIVE**

1. To build a system that can receive input and generate automatically output in easy way and short time .
2. To build a monitoring system that is able to monitor and manage all library operations efficiently
3. Give an opportunity to librarians to reduce mistakes that always happen during manual method.
4. To store properly the library items in order to maintain their security.
5. To enter and preserve details of the various issues and keep a track on their returns.

* **OUTCOME**

Library Management System is an application created using c programming. This is a simple project compiled in Code Blocks IDE using GCC compiler. It details the type of books, the list of books, etc. Only a person with the login credentials can access the Library Management System. That person can perform many operations like adding the book details, removing the book details, displaying the book details, modifying the book details, etc.

FEATURE OF THIS SYSTEM

* Add Books
* Delete Books
* View Books
* Search Books
* Close Application
* **PROJECT DIAGRAM**

Book ID: Book Name: Author Name:

Book Found

Delete Successfully

Added Successfully

Book ID: 

Book ID:

Book ID:

Book ID: Book Name: Author Name:

Add Book

Search Book

Close Application

Delete Book

View Book

MENU

* **IMPLEMENTATION PROCEDURE**

1. The code defines a structure called `Book` that represents a book's information, including an ID, name, and author.

2. It declares an array of “Book” structures called “books” with a maximum size of `MAX\_BOOKS` and initializes the variable `numBooks` to 0.

3. It defines several function prototypes: `addBook()`, `deleteBook()`, `viewBooks()`, and `searchBook()`.

4. In the `main()` function, a menu is displayed to the user, presenting various options: adding books, deleting books, viewing books, searching for a book, or closing the application.

5. Inside a `while` loop, the user is prompted to enter their choice. The input is stored in the `choice` variable.

6. Depending on the user's choice, the corresponding function is called to perform the requested action.

7. Declaring a struct `Book` with attributes such as ID, Name and Author

8. Declaring a global array `books` to store the books.

9. Declaring a global variable `numBooks` to keep track of the number of books in the array.

10. Implementing functions like addBook()` delete book(), display book() to add , delete,and displaying a new book to the `books array.

11. In the main function:

a. Initialize `numBooks` to 0.

b. Use a loop to display a menu of options to the user (add, delete, display, exit).

c. Read the user's choice and perform the corresponding action.

d. Repeat the loop until the user chooses to exit.

12. Test the program by adding, deleting, and displaying books.

**CODE OF THE SYSTEM**

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <string.h>**

**#define MAX\_BOOKS 100**

**typedef struct {**

**int id;**

**char name[50];**

**char author[50];**

**} Book;**

**Book books[MAX\_BOOKS];**

**int numBooks = 0;**

**int addBook();**

**int deleteBook();**

**void viewBooks();**

**int searchBook(int id);**

**int main()**

**{**

**int choice;**

**printf("\n\t\t\t\t\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* MAIN MENU \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");**

**printf("\n\t\t\t\t\t\t1. Add Books");**

**printf("\n\t\t\t\t\t\t2. Delete Books");**

**printf("\n\t\t\t\t\t\t3. View Books");**

**printf("\n\t\t\t\t\t\t4. Search Book");**

**printf("\n\t\t\t\t\t\t5. Close Application");**

**printf("\n\t\t\t\t\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");**

**while (1) {**

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

**scanf("%d", &choice);**

**switch (choice) {**

**case 1:**

**if (addBook()) {**

**printf("Book added successfully.\n");**

**}**

**break;**

**case 2:**

**if (deleteBook()) {**

**printf("Book deleted successfully.\n");**

**}**

**break;**

**case 3:**

**viewBooks();**

**break;**

**case 4: {**

**int id;**

**printf("Enter Book ID to search: ");**

**scanf("%d", &id);**

**if (searchBook(id)) {**

**printf("Book found!\n");**

**} else {**

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

**}**

**break;**

**}**

**case 5:**

**printf("Closing application.\n");**

**exit(0);**

**default:**

**printf("Invalid input\n");**

**break;**

**}**

**}**

**return 0;**

**}**

**int addBook() {**

**int id;**

**char name[50];**

**char author[50];**

**if (numBooks == MAX\_BOOKS) {**

**printf("Cannot add more books. Library is full.\n");**

**return 0; // Unsuccessful**

**}**

**printf("Book ID: ");**

**scanf("%d", &id);**

**// Check if the book with given ID already exists**

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

**if (books[i].id == id) {**

**printf("Book with ID %d already exists.\n", id);**

**return 0; // Unsuccessful**

**}**

**}**

**printf("Book Name: ");**

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

**printf("Author Name: ");**

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

**books[numBooks].id = id;**

**strcpy(books[numBooks].name, name);**

**strcpy(books[numBooks].author, author);**

**numBooks++;**

**return 1; // Successful**

**}**

**int deleteBook() {**

**int id;**

**printf("Enter Book ID to delete: ");**

**scanf("%d", &id);**

**int found = 0;**

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

**if (books[i].id == id) {**

**found = 1;**

**for (int j = i; j < numBooks - 1; j++) {**

**books[j] = books[j + 1];**

**}**

**numBooks--;**

**return 1; // Successful**

**}**

**}**

**if (!found) {**

**printf("Book with ID %d not found.\n", id);**

**return 0; // Unsuccessful**

**}**

**return 0; // Unsuccessful**

**}**

**void viewBooks() {**

**if (numBooks == 0) {**

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

**return;**

**}**

**printf("\n%-10s %-30s %-30s\n", "Book ID", "Name", "Author");**

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

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

**printf("%-10d %-30s %-30s\n", books[i].id, books[i].name, books[i].author); } }**

**int searchBook(int id) {**

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

**if (books[i].id == id) {**

**printf("\n%-10s %-30s %-30s\n", "Book ID", "Name", "Author");**

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

**printf("%-10d %-30s %-30s\n", books[i].id, books[i].name, books[i].author);**

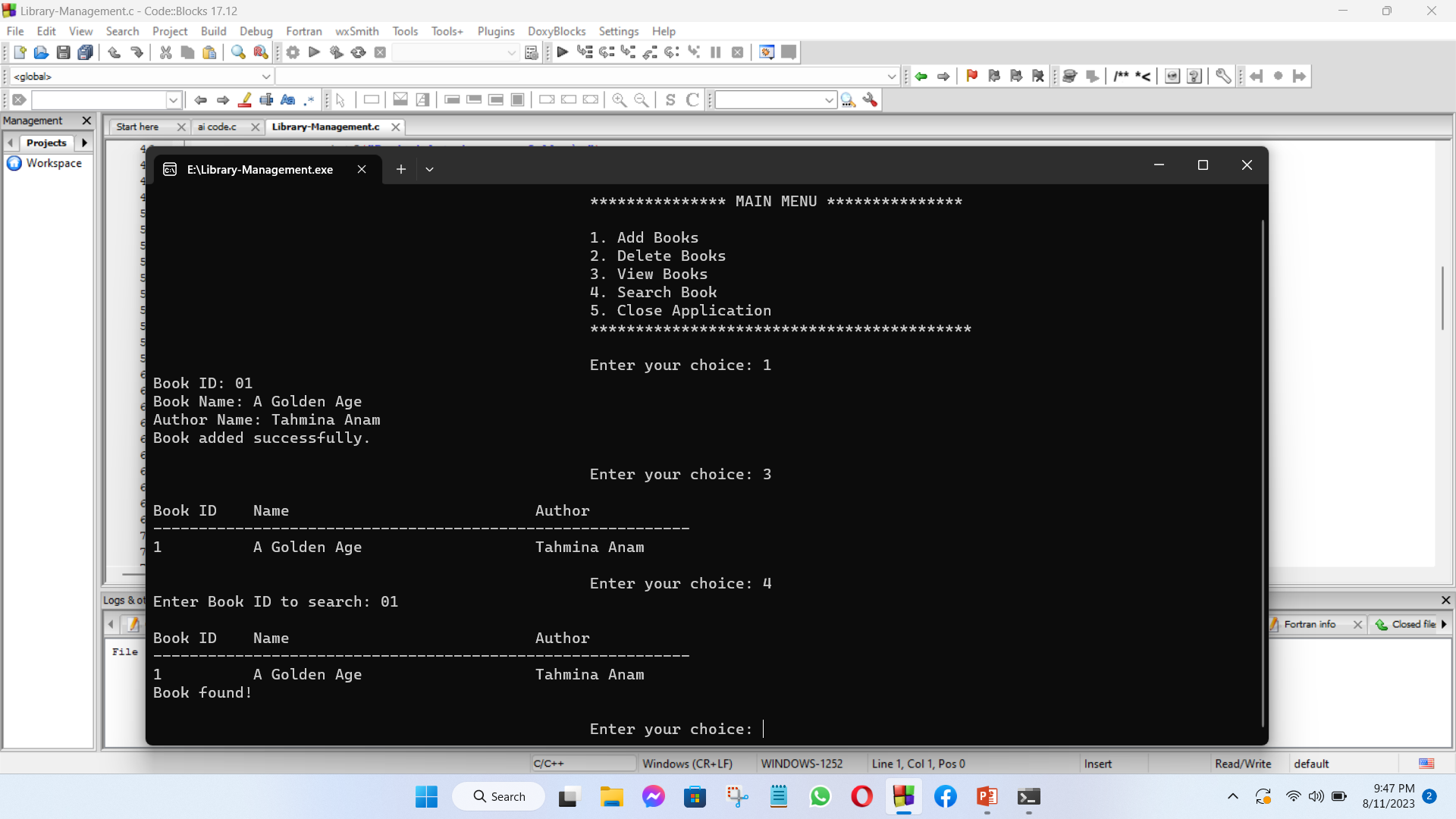
**return 1; // Book found } }**

**return 0; // Book not found }**

* **DISCUSSION AND RESULT**

Adding new book is one of the most straightforward features of this project. We just need to take the information of book’s from the user as input and store it.

RESULT

****

* **CONCLUSION**

The Library Management System allows the user to store the book details and the person's details. This software allows storing the details of all the data related to library.