Library Book Management System in C

1. Program Design Overview

The program will be menu-driven, operating in a continuous loop until the user chooses to exit. Each menu option corresponds to a function handling one of the core operations.

Main Menu Example

====== Library Book Management System ======
1. Add New Book
2. Search Book
3. Update Book Record
4. Delete Book Record
5. Display All Books
6. Exit
==========
Enter your choice: _

2. Data Structure Design

Each book will be represented as a structure.

Structure Definition:
#define TITLE_LEN 100 #define AUTHOR_LEN 100 #define ISBN_LEN 20 typedef struct { int id; char title[TITLE_LEN]; char author[AUTHOR_LEN]; char isbn[ISBN_LEN]; int available; } Book;

3. File Design

All book records will be saved in a file named **library_records.txt**.
Example record format:
1|The Alchemist|Paulo Coelho|9780061122415|1 2|C Programming Absolute Beginner's Guide|Greg Perry|9780789751980|0

4. Function Design

Key functions:
- void addBook();
- void searchBook();
- void updateBook();
- void deleteBook();
- void displayBooks();
- void saveBookToFile(Book book);
- int loadBooks(Book **books);
- void saveAllBooks(Book *books, int count);

5. Memory Management Plan

Dynamic memory allocation will be used:
Book *books = NULL; int count = 0; Using malloc(), realloc(), and free() appropriately.

6. File Handling Strategy

Different modes:
- Append mode ("a") to add books
- Read mode ("r") to search or display
- Write mode ("w") to update/delete
Use fprintf() and fscanf() or fgets() with sscanf().

7. Example Data Flow (Add Book)

1. User selects Add New Book
2. Program asks for book details
3. Data stored in Book struct
4. Appended to library_records.txt
5. Confirmation message displayed

8. Example Data Flow (Search Book)

1. User selects Search Book
2. Program asks for keyword
3. Loads each record and compares using strstr()
4. Displays matching results