Library Management System

A Comprehensive Guide

Team Securo

2025-01-03

Table of contents

1 Introduction

The Library Management System (LMS) is designed to efficiently manage books, members, and borrowing records. It is modular, scalable, and focused on functionality and usability, adhering to project guidelines and constraints. This system simulates library operations while ensuring clarity and maintainability through structured programming.

---

2 Team Members

- **Team Leader**: Mahmoud Zaki (24-101324)
- **Team Members**:
  - Ahmed Amir (24-101432)
  - Ahmed El Gayyar (24-101388)
  - Ahmed Essam (24-101175)
  - Abdelrahman Ghazy (24-101264)
  - Ziad Ahmed (24-101036)

---

3 Memory Layout

3.1 Constants

- `MAX_BOOKS = 50`: Maximum number of books in the library.
- `MAX_MEMBERS = 30`: Maximum number of library members.
- `MAX_BORROWED_BOOKS = MAX_MEMBERS * 5`: Maximum number of books that can be borrowed.
- `DATE_LENGTH = 11`: Length of the date string (DD/MM/YYYY + null terminator).

3.2 Data Structures

3.2.1 Books Data

Tracks books by their unique IDs and available copies.

Field	Description
`books_data[i][0]`	Book ID
`books_data[i][1]`	Number of available copies

3.2.2 Members Data

Stores library member IDs.

Field	Description
`members_id[i]`	Member ID

3.2.3 Borrowed Books Data

Tracks borrowed book IDs and corresponding member IDs.

Field	Description
`borrowed_books_data[i][0]`	Book ID
`borrowed_books_data[i][1]`	Borrower ID
`borrowed_books_date[i]`	Borrow date (DD/MM/YYYY)

3.3 Data Counters

Field	Description
`data_count[0]`	Total distinct books
`data_count[1]`	Total copies of all books
`data_count[2]`	Total library members
`data_count[3]`	Total borrowed books

4 Functions and Handlers

4.1 Helper Functions

4.1.1 `get_books_data`

Processes book data and calculates total copies.

Parameters: - `books_data`: Array for storing book IDs and copies. - `books_count`: Pointer to store the number of books.

Returns: - Total number of book copies.

4.1.2 `get_members_data`

Processes member IDs.

Parameters: - `member_id`: Array for storing member IDs.

Returns: - Total number of members.

4.1.3 `get_borrowed_data`

Processes borrowed book data and dates.

Parameters: - `borrowed_books_data`: Array for storing borrowed book IDs and member IDs. - `borrowed_books_date`: Array for storing borrow dates.

Returns: - Total borrowed book entries.

4.1.4 Command-Specific Functions

Function	Description
`number_books`	Outputs total number of books.
`number_members`	Outputs total number of members.
`book_id_min`	Outputs the smallest book ID.
`books_available`	Lists books with available copies.
`list_book_borrowers`	Lists borrowers of a specific book.
`list_member_books`	Lists books borrowed by a specific member.
`most_borrowed`	Outputs the most borrowed book ID.
`members_less`	Lists members who borrowed fewer than `n`.
`books_unborrowed`	Lists books that have not been borrowed.

Function	Description
`books_borrowed_days`	Counts unique borrowing days.
`books_per_member`	Lists members and their borrow counts.
`overlapping_borrowers`	Lists borrowers who borrowed the same book on the same day.

4.2 Handlers

4.2.1 `get_data_handler`

Manages data collection from the user and calculates initial statistics.

Outputs: - Populates `data_count` with counts for books, members, and borrowed books.

4.2.2 `commands_handler`

Processes user commands and invokes relevant functions.

Supported Commands: - `number_books` - `number_members` - `book_id_min` - `books_available` - `list_book_borrowers [ID]` - `list_member_books [ID]` - `most_borrowed` - `members_less [n]` - `books_unborrowed` - `books_borrowed_days` - `books_per_member` - `overlapping_borrowers [ID]` - `quit`

---

5 Implementation Details

5.0.1 Libraries Used

Library	Purpose
`stdio.h`	Input/output functions (`printf`, `scanf`).
`stdlib.h`	Conversion functions (`atoi`).
`string.h`	String comparison (`strcmp`).
`ctype.h`	Character manipulation (`tolower`, `isspace`).

5.0.2 Debugging

- Comments indicate locations where debug `printf` statements were used.
- Debugging statements are disabled per project guidelines.

---

6 Sample Input and Output

6.1 Input

```
Books:
101 5
102 3
103 7
Members:
201000
202000
Borrowed_Books:
101 201000 01/12/2024
102 202000 03/12/2024
```

6.2 Output

```
Number_Books
3
Number_Members
2
Books_Available
101
102
Most_Borrowed
101
Quit
Thanks!
```

---

7 Future Enhancements

1. **Dynamic Memory Allocation**: Replace static arrays with dynamic structures.
2. **Improved Date Handling**: Use libraries for robust date validation and processing.
3. **Enhanced User Interface**: Implement a menu-driven CLI for improved usability.
4. **Advanced Reporting**: Add analytics and visualizations for borrowing trends.

This document provides an in-depth guide to the Library Management System, including its design, implementation, and functionality, in compliance with project guidelines.