**Open Ended Lab -1**

**Spring 2025**

**Course Title: Data Structures Lab**

**Course Code:** **CSE1302**

**Section: 4**

**Submitted by:**

**Student Name: Md Abdullah Al - Hameem**

**ID: 242014036**

**Department of CSE**

**University of Liberal Arts Bangladesh (ULAB)**

**Library Management System**

This lab report covers the design and development of a basic library management system built using the C programming language. It relies on fundamental data structures like arrays to store book information and includes essential algorithms such as insertion sort for organizing records and binary search for quick search. Additionally, the program uses file system to save and retrieve data, allowing the system to keep records between sessions and make the program more dynamic. The system is designed to manage a collection of books using an array of structures. The key concepts used in the programming are:

* Adding new books.
* Removing existing books.
* Displaying all books in a sorted order.
* Searching for a book by its ID using binary search.

A menu-driven interface allows the user to perform these operations continuously until they decide to exit.

Use of Structures:

The program uses a struct named library to group related attributes:

* id: A unique integer identifier for each book.
* title: A character array to store the title of the book.
* author: A character array to store the author’s name.

This provides a clear and organized way to represent a single book’s data and allows operations on multiple book records using arrays of the structure.

Array of Structures

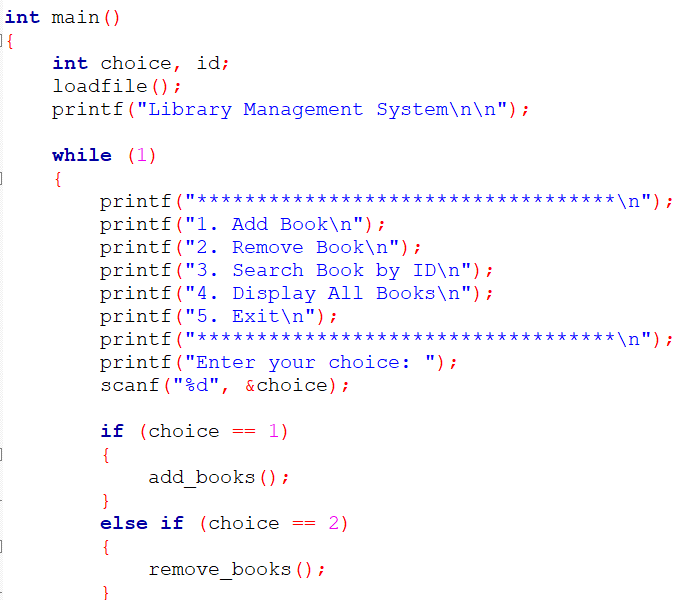
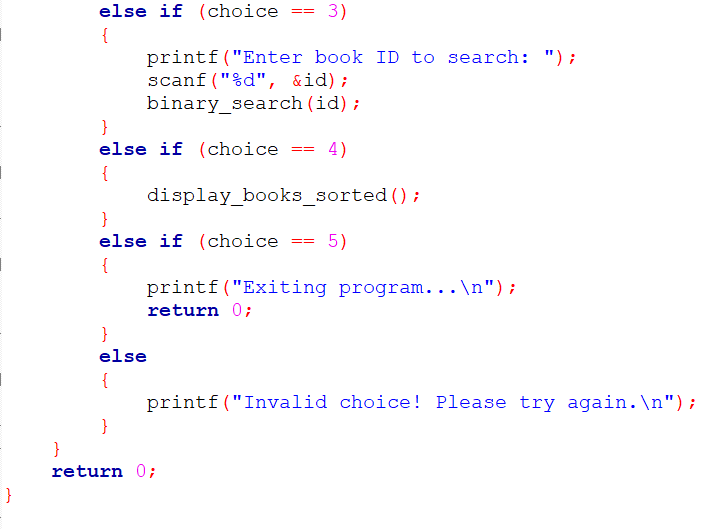
An array of structures named books (of size 100) is used to store multiple books records. It allows user to store multiple book details in a single variable. It shortens the code ensuring better user readability.

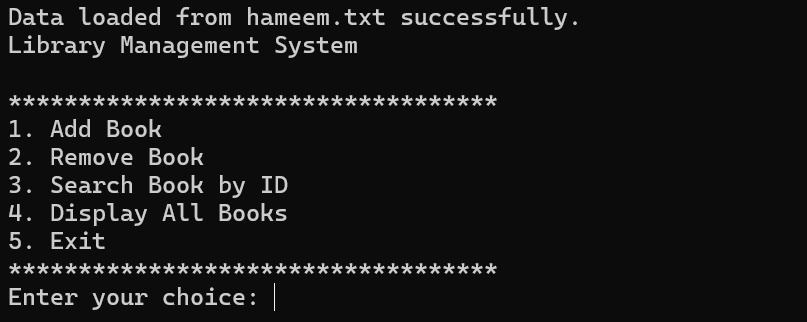
* Purpose: Acts as the primary storage for the library's records.
* Operations: Books are added, deleted, and searched within this array.
* Index Management: A global variable n keeps track of the number of books currently stored.

Array Operations in the System:

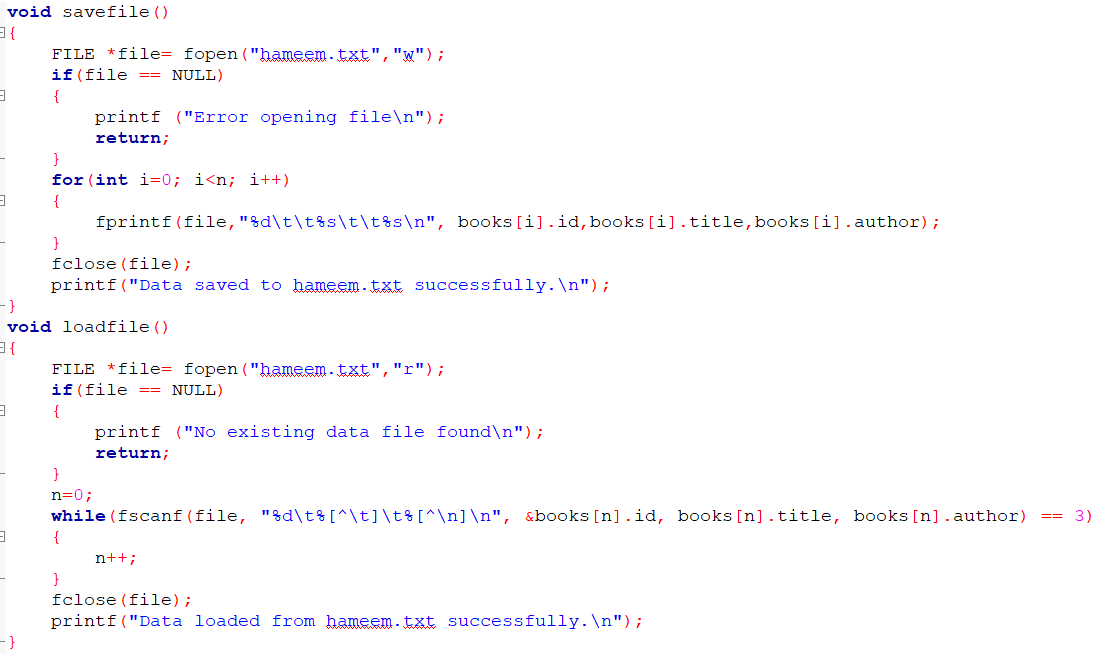
The program is divided into multiple functions, each responsible for a specific task:

• main(): Manages the overall program flow and user input. The main function pops the menu driven input and asks the user for a choice. The it calls the function based in the choice of the user. Once the task is done it can also exit the program.

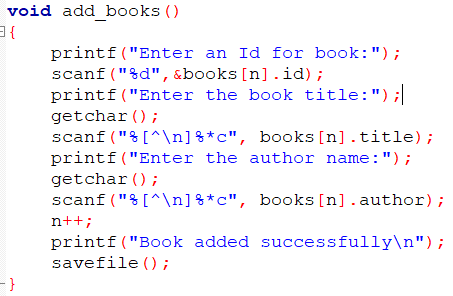
 

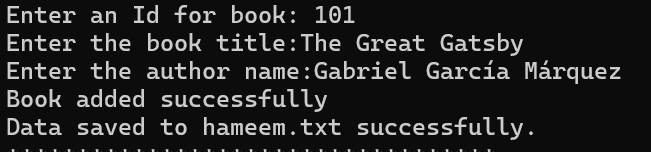


• savefile() and loadfile(): Handles saving data to a file and loading data from a file. The Program saves all data to a notepad file named “Hameem.txt” and this allows data to be stored permanently in a file, so it remains available even after the program ends. This system reduces the need for constant user input during program execution and can load previously saved data from the user. Data stored in files reduces the reliance on memory during program execution, making programs more efficient.

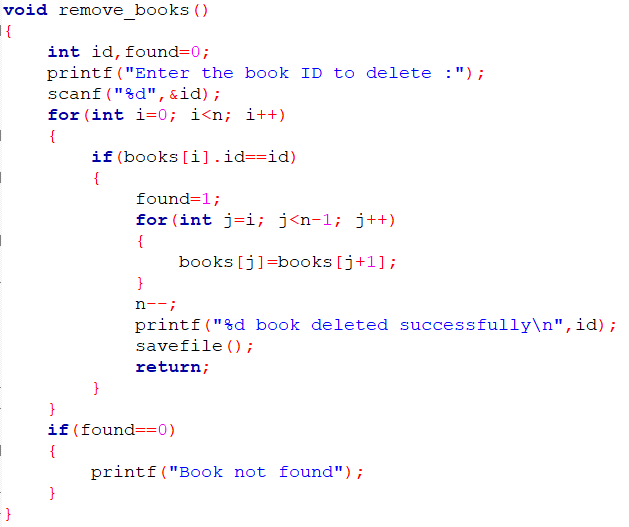


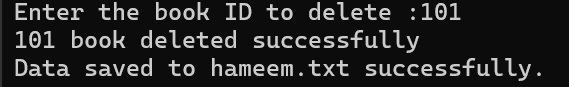
• add\_books(): Adds a new book in the library record. Asks the user the user to input the book id, book name and author name stores it in the function savefile() . The input is taken using a menu driven user interface.





• remove \_books(): Deletes a book record. The user is prompted for the book ID. The system searches the array for the matching ID. Once found, the array elements are shifted left to remove the gap. Then n is decremented and the file is updated.

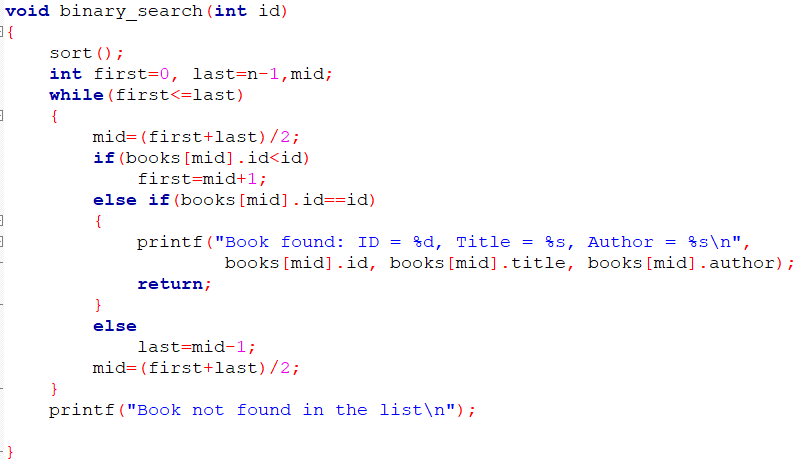
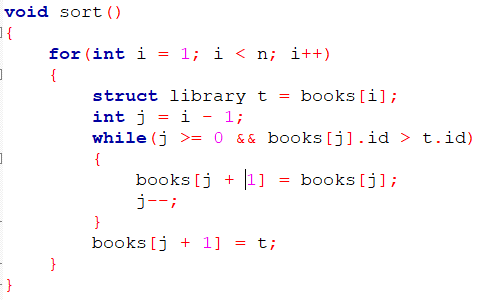




• binary\_search(int id): Binary search algorithm is used here to efficiently locate a book by its ID. At first the system uses an insertion sort algorithm to sort the books array by the book's ID before performing searches or displaying the books, since binary search requires sorted data. This algorithm:

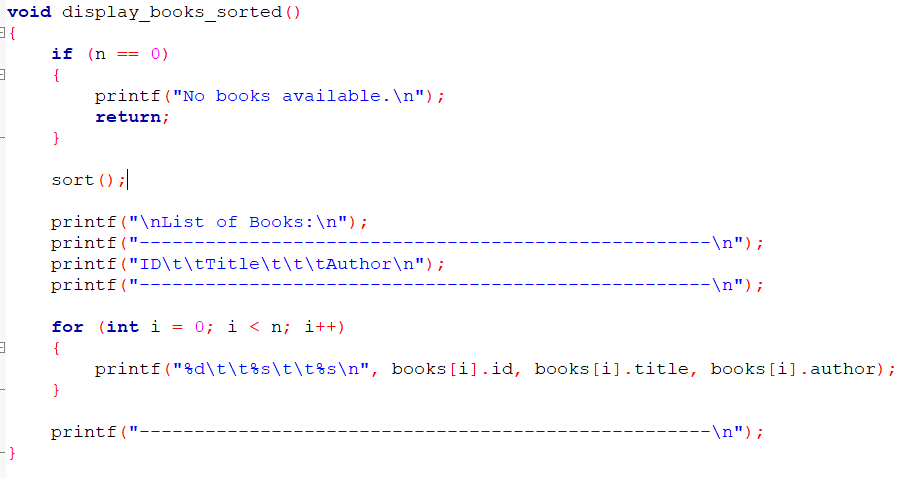
* Divides the sorted array into halves.
* Compares the target ID with the middle element.
* Repeats the process on the appropriate half until the target is found or the search interval is empty.

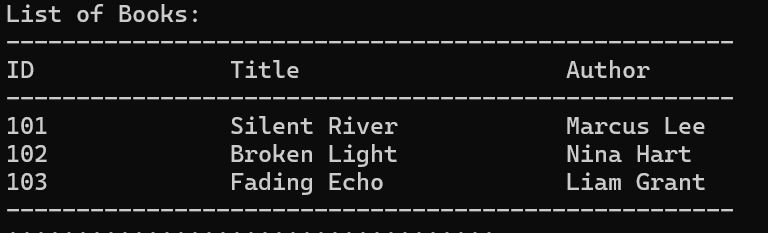
Binary search has a time complexity of O(log n), making it an efficient search method for large datasets.



• display\_books\_sorted(): This function displays all the books in a sorted order. The books are sorted in ascending. And then is display the list along with the book id, title and author name.





Conclusion:

The Library Management System demonstrates the effective use of arrays to store structured data, alongside essential operations such as insertion, deletion, sorting, and searching as well as viewing all records in a formatted manner. The integration of file system for saving data guarantees that data remains intact even after the program is closed. It demonstrates essential programming skills, including the use of structures, file handling, and menu-driven interfaces. This assignment serves as a practical introduction to managing data in real-world applications.