National University of Computer and Emerging Sciences, Lahore Campus

Programming Fundamentals

Term Project

Section: BSE-1A, 1B **Due Date:** 27th November 2022

Library Management System

You are required to implement a **Library Management System** which will provide the following features:

Features

1. A Login or Sign-Up functionality. The system should ask the user to enter his credentials in order to use the features of the system. If the user does not exist, the system should display a proper msg and ask the user to sign up instead. The system should only let a registered user access its functionalities.

2. A menu that will provide access to the features of the Library Management System. Note that the system should only terminate when the user instructs to do so. Otherwise, after execution of each function the system should ask the user to return to the menu or execute the functionality again.

3. The storage of Books. A book should have a title, ISBN number, author, date of publishing, quantity available in the library.

4. The retrieval of books stored in the database by searching. For this purpose, the user can input the title of the book or its ISBN number.

5. Adding new books into the library database and deleting a book from the database. The database will in the form of text files on the local storage.

6. Lending a book to a person. Returning an issued book to the library.

Requirements

Adhere to the following instructions when implementing the system.

1. You need to create Structs of Book and Lending. A book should have information such as title, ISBN number, author, date of publishing, quantity available and total quantity in the library. Lending will have information like issued Book (assign the Book Struct to this), Issuing Date, Date of return, name of the person to whom the book has been issued and fine applicable for late return.

2. All of the lending and book structs should be stored in their own file (txt) in their respective directories. For example, the lending should be stored in the Lending folder and the Books should be stored in the Books folder. You can access all the files in a directory using the readdir() function.

3. Assume that the total capacity of the library for storing books is 10,000. You need to maintain two arrays of structs in order to store the data of Books and Lendings in the memory. The data should be retrieved from the txt files upon the start of the program, when a book is added and deleted.

4. Any change in the memory should also reflect in the files stored in the disk, so that the data integrity is maintained.

5. When a book is issued to a person, you need to decrease the quantity of the books *available* in the library. If the quantity of a book has reached zero, you must not let the book be issued to a person. The *total* quantity of the books should also decrease when a book is deleted from the library records.

6. The system should keep track of the fines collected from late returns. Whenever a person returns a book, the system should ask the user if the person is returning the book late or not, if it is a late return then fined amount should be added to the library's treasury. You can maintain a variable for this purpose.

7. All of the data of the system should be stored on files so that the data is not lost when the system is closed. The system should load or reload the data automatically whenever the system is started or any change is made to the database.

8. The book searching of the system should be efficient. (Hint: Sorting & Binary search)

9. Bonus marks will be given to the groups who implement a Graphical User Interface.

10. You must implement this project only on Visual Studio Community Edition.

Additional Notes

For this project, you can use the C++ string library for ease. Other than that, you must not use any other STL library (including algorithm). You are not allowed to use OOP concept (classes) and dynamic memory allocation (pointers, variable length arrays).

The instructions given are abstract so that the groups can think up their own logic and chances of coincidental plagiarism are miniscule. Therefore, any plagiarism from online resources or other students will result in a direct F grade and DC. Cross-sectional plagiarism will be checked in order to ensure originality of code.

Good luck!