UNIVERSITY OF TECHNOLOGYJAMAICA
FACULTY OF ENGINEERING AND COMPUTING
Data Structures - Project (20%)

Instructions

- This is a group project, no individual work, 3-5 students per group
- Penalties for plagiarism will be ***FULLY ENFORCED.***

Due Date: Week of March 24, 2025. No extension will be allowed

Project Title: Library Management System

Project Overview

You will design and implement a "Library Management System" that allows for efficient management of books, patrons, and library operations using various data structures, including stacks, queues, linked lists, and binary trees. The system should provide functionalities like book check-in/checkout, search functionalities, and user-friendly interactions.

Project Requirements

Functional Requirements

1. Book Management:

Implement a linked list to manage a collection of books. Each book should have attributes such as title, author, ISBN, and availability status.
Implement a binary search tree (BST) for efficient searching and sorting of books by title or author.

2. Patron Management:

Implement a linked list to manage patrons. Each patron should have attributes such as name, library card number, and the list of books currently checked out.
Use a queue data structure to handle checkouts and returns of books, ensuring that the first patron to check out a book is the first allowed to return it.

3. Check-In/Check-Out System:

Implement functions for checking books in and out. When checking out a book, the system should utilize a stack to hold the actions for undoing a checkout.
Ensure that the system manages book availability correctly.

4. Search Functionality:

Implement search functionality to find books by title, author, or ISBN using your BST.

5. User Interface:

Create a command-line/GUI interface that allows users to interact with the system – adding books, checking out books, checking in books, searching for books, and viewing patron details.

6. Statistics:

Track and display libraries' statistics, including total books, total patrons, and the number of current checkouts.

7. File System

Implementation of file system to store data on book management, patron management and passwords.

8. Password Management

The system should have a default password for admin. The user name is admin and the password is admin. A user can be either an admin or a user. When registering to the system, the system will generate and display a default password that must be changed on first login. A mechanism must be in place i.e. hashing of password so that no clear text password is stored in the password file.
Password must be able to be changed by a user at anytime.

Implementation Details

- Choose a programming language (Java or C++)
- Create classes for Book, Patron, Password and the Library Management System.
- Utilize appropriate data structures as specified.
- Modularize your code to maintain readability and facilitate testing.
- Include error handling and edge cases (e.g., checking out an already checked-out book, searching for non-existent books, checking username and password).

Mark Scheme

Criteria	Description	Marks Available
Functionality	System works as specified (book management, patron management, check-in/out functionalities, search)	35
Data Structure Implementation	Appropriate usage and implementation of linked lists, stacks, queues, and binary trees	20
Code Quality/User Manual	Clean, readable code with proper naming conventions, comments, and user manual (with photos/screen shots)	20
User Interface	User-friendly command-line/GUI interface with clear prompts and error handling	15
Error Handling	Robust error handling for invalid operations and edge cases	5
Demo	Demonstrating system functionalities	25
File System	Implementation of files for password, book management, patron and library system	15
Password	Implementation of password system	15
Individual Report	Accurate description of work done by group member	10
Total		150

Submission Guidelines

- Submit your code in a compressed format (ZIP or TAR).
- Include also the USER MANUAL file with instructions on how to compile/run your project.

- Lastly include a brief report (1 page) individual report that describes your contribution, data structures used, challenges faced, and how you address.