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 November 18, 2024 . No extension will be allowed

Project Title: Course Registration System

Project Overview

The objective of this project is to create a simplified Course Registration System for ABC University using key data structures: queues, stacks, and linked lists. The system should allow students to register for courses, check enrollment status, manage course capacities, and manage waitlists. This project will allow you to get a better understanding of data structures by applying them in a real-world scenario.

Project Requirements

Data Structures:

Linked List: Use linked lists to store information about students and courses, allowing for dynamic resizing and efficient insertions/deletions.
Stacks: Implement a stack to manage the history of student course registrations, allowing students to undo their last registration if needed.
Queues: Use a queue to manage a waitlist for courses that have reached capacity.
Functionality:

Student Management:

Add, remove, or modify a student record (linked list).
Each student should have attributes like ID, name, and a list of enrolled courses (linked list).

Course Management:

Add, remove, or modify a course record (linked list).
Each course should have attributes like course code, title, credits, maximum capacity, and a list of enrolled students (linked list).

Course Registration:

Allow students to register for courses, checking for prerequisites and capacity.
If a course is full, add the student to the waitlist (queue) or if the student is needs to do the course add the student to the priority (queue).

Undo Registration:

Allow a student to undo their last registration action using a stack.
Update student record reflecting the undo.
Update the course capacity after each course undo registration
Please note that if a student undo a registration for a course that has a priority queue or a waitlist queue then the first person in the priority queue registration should be updated if no priority exist then the first person in the waitlist queue registration should be updated.

Display Functions:

Display all courses available for registration.
Display all students registered in a specific course.
Display waitlists for courses that are full.

User Interface:

Create a simple command-line interface (CLI) that allows users (students) to interact with the system.
Provide clear options for each functionality and make sure to handle invalid inputs gracefully.

Error Handling:

Implement error handling for cases such as:
Trying to enroll in a course with unmet prerequisites.
Selecting a non-existent course or student.
Handling full courses and managing the overflow with waitlists.

Suggested Technologies

Programming Language: Java or C++.

Project Deliverables

a. Source Code:

Structure the code into modules for clarity.
Document the code with comments explaining the functionality of each section.

b. User Manual:

Provide instructions on how to run the application and use its features.

c. Presentation:

Demonstrating the system and discussing the implementation details.

Timeline

Week 1: Project kickoff, requirement gathering, and initial design.
Week 2: Implement linked list for student and course management. Implement stack for undo functionality and queue for waitlist management.
Week 3: Develop the command-line interface and integrate all components. Testing and debugging.
Week 4: Final touches and presentation preparation.

Assessment Criteria (Total 150 marks)

Objective	Description	Mark
Student Management:	Add, remove, or modify a student record (linked list). Each student should have attributes like ID, name, and a list of enrolled courses (linked list).	25
Course Management:	Add, remove, or modify a course record (linked list). Each course should have attributes like course code, title, credits, maximum capacity, and a list of enrolled students (linked list).	25
Course Registration:	Allow students to register for courses, checking for prerequisites and capacity. If a course is full, add the student to the waitlist (queue) or if the student is needs to do the course add the student to the priority (queue).	25
Display Functions:	Display all courses available for registration. Display all students registered in a specific course. Display waitlists for courses that are full.	15
Undo Registration	Allow a student to undo their last registration action	15
Error Handling	Implement error handling for cases which will cause program crashing or unexpected results.	10

Usability	Ease of use and clarity of the command-line interface.	15
Code Quality	Clean, well-structured, and properly documented code.	5
User Manual	Provide instructions on how to run the application and use its features.	15