Faculty of Engineering, University of Jaffna,

Department of Computer Engineering.

EC5080: Software Construction

Lab 03

Duration: 3 Hours **Lecturer**: Ms. Sujanthika M.

Defining Linked Structures.

Objectives:

- Build on their knowledge of classes and objects.
- Implement linked structures like trees and graphs.
- Understand how objects can be linked dynamically.
- Explore tree and graph traversal techniques.

In this lab, you will extend the Library Management System you implemented in Lab 2 to include a catalog system using trees and graphs.

Part 1:

1. Create a `BookNode` class representing a node in the BST.
2. Implement a `BookCatalog` class with the following:
    a. `insert(Book book)`: Adds a book to the BST.
    b. `search(String title)`: Searches for a book by title.
    c. `inOrderTraversal()`: Displays books in sorted order.

Part 2:

1. Create a `Graph` class using an adjacency list.
2. Implement methods to:
    a. Add a book recommendation connection.
    b. Display all recommendations for a given book.
3. Demonstrate graph traversal (DFS or BFS) to find recommendations.

Part 3:

1. Demonstrate how new books are dynamically added to BST and graph.
2. Implement a method to remove a book from BST and update the graph.
3. Observe and comment on memory allocation and deallocation.

Part 4:

1. Execute all implemented methods.
2. Compare BST and graph performance for searching and retrieving books.
3. Discuss findings in comments.

Submission Guidelines:

- Submit a well-structured Java source file with comments
- Provide screenshots for the outputs.
- Provide correct names for your code files and outputs
- Any plagiarized work will be given zero marks
- Late submissions are not allowed

Section	Marking Criteria	Allocated Marks
Part 1	Correct BST node and structure	10
	`insert()` and `search()` implementation	10
	`inOrderTraversal()` method for sorted display	5
Part 2	Correct graph representation using adjacency list	10
	Book recommendation connections implemented correctly	10
	Graph traversal (DFS/BFS) correctly applied	5
Part 3	Correct dynamic memory allocation for linked structures	10
	Removal of books from BST and graph updates	10
	Documentation of memory observations	5
Part 4	Correct execution of all methods	10
	Performance comparison documented	10
	Well-structured and commented code	5