**Aror University of Art, Architecture, Design &** 

**Heritage Sukkur**

Department of AI-Multimedia and Gaming

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Lab 05: Fundamentals of Stack using Array and LinkedList Date: 24 Sep, 2024 Subject: Data Structure (CSC221), Fall 2024 Instructor: Abdul Ghafoor

**Lab objectives:** The goal of this lab is to help students understand the **stack** data structure by implementing it using both **array** and **linked list** in Java. Students are required to implement basic stack operations such as **push**, **pop**, **peek**, **isEmpty**, and **size** using both data structures.

**Task 01: Implement a stack using an array with the following methods:**

● push(int data): Adds an element to the top of the stack.

● pop(): Removes and returns the top element from the stack.

● peek(): Returns the top element without removing it.

● isEmpty(): Checks if the stack is empty.

● size(): Returns the number of elements in the stack.

**Instructions**:

● Define a fixed capacity for your array (for example 5).

● Handle edge cases such as **stack overflow** (when the array is full) and **stack underflow** (when trying to pop from an empty stack).

● Write a **main method** to test all the stack operations by pushing and popping elements, and printing the stack after each operation.

**Part 02: Stack Implementation Using a Linked List**

1. **Task**: Implement a **stack** using a **singly linked list** with the following methods: ○ push(int data): Adds an element to the top of the stack.

○ pop(): Removes and returns the top element from the stack.

○ peek(): Returns the top element without removing it.

○ isEmpty(): Checks if the stack is empty.

○ size(): Returns the number of elements in the stack.

2. **Instructions**:

○ Use a **linked list** where each node contains an integer data field and a pointer to the next node.

○ Handle edge cases such as trying to **pop from an empty stack**.

○ Write a **main method** to test all the stack operations by pushing and popping elements, and printing the stack after each operation.

**Part 03: Leetcode Examples for stack**

https://leetcode.com/problems/valid-parentheses/description/?envType=problem-list-v2&envId=stack

|  |
| --- |

https://leetcode.com/problems/palindrome-linked-list/description/?envType=problem-list-v2&envId=stack

|  |
| --- |

https://leetcode.com/problems/next-greater-element-i/description/?envType=problem-list-v2&envId=stack

|  |
| --- |

://leetcode.com/problems/final-prices-with-a-special-discount-in-a-shop/description/?envType=problem-list v2&envId=stackhttps