COMSATS University Islamabad, Abbottabad Campus
Department of Computer Science

LAB #9

Data Structures

Class: BCS- 4C & D

Q1. Implement a **Double Ended Queue (DEQueue)** using an array.

Perform the following operations

- insertFront()
- insertRear()
- deleteFront()
- deleteRear()
- isEmpty()
- isFull()
- display()

Implement above methods in **DEQueue using a circular array**. Show how the front and rear references/variables update in each operation.

Q2. Perform the following operations on an initially empty DEQueue of size 4:

F, R = −1 initially

Operations:

1. insertFront(7)
2. insertRear(8)
3. insertRear(9)
4. deleteFront()
5. insertFront(6)
6. deleteRear()

Display the final queue and pointer positions.

Q3. **Implement a Priority Queue using an array** where:

- Each element has:

  TaskName, **PriorityNumber**
- Insert elements based on priority (lower number = higher priority).
- Display all elements.

Q4. Implement a **Priority Queue for Hospital Patients**, where each patient record stores:

- PatientID
- Name
- SeverityLevel

Operations to implement:

- Enqueue (higher severity → higher priority)
- Dequeue
- Display all patients
- Count total patients

Q5. Emergency Alert System:

Messages have priority levels (1 = high).

Simulate:

- Insert an alert
- Process highest priority alert
- Display message queue