Data Structures Lab Manual

# Lab 5: Queue and Circular Queue (Using Arrays and Linked Lists)

Prepared for: Students of Data Structures

Department of Computer Science  
Fast School of Computing

# Objectives

1. To understand the concept of Queue and Circular Queue.  
2. To implement Queue using Arrays and Linked Lists.  
3. To implement Circular Queue using Arrays and Linked Lists.  
4. To analyze the advantages and limitations of different implementations.  
5. To develop problem-solving and coding skills in handling queue operations.

# Lab Outcomes

After completing this lab, students will be able to:  
1. Differentiate between simple Queue and Circular Queue.  
2. Implement Queue using Array and Linked List in C++.  
3. Implement Circular Queue using Array and Linked List in C++.  
4. Perform enqueue and dequeue operations in both linear and circular queues.  
5. Apply queue concepts to solve real-world problems.

# Lab Task

Task 1: Implement a simple Queue using Arrays.  
 - Write a program to perform enqueue, dequeue, and display operations.  
 - Handle queue overflow and underflow conditions.

Task 2: Implement a Queue using Linked List.  
 - Perform enqueue and dequeue operations.  
 - Display the elements of the queue.

Task 3: Implement a Circular Queue using Arrays.  
 - Write a program to perform enqueue, dequeue, and display operations.  
 - Demonstrate how the rear pointer wraps around.

Task 4: Implement a Circular Queue using Linked List.  
 - Perform enqueue, dequeue, and display operations.  
 - Demonstrate circular linkage in the linked list implementation.

Task 5: Create a menu-driven program that allows the user to choose between:  
 1. Queue using Array  
 2. Queue using Linked List  
 3. Circular Queue using Array  
 4. Circular Queue using Linked List  
The program should allow enqueue, dequeue, and display operations in each case.

# Submission Guidelines

- Submit your .cpp file with proper comments.  
- Make sure your program compiles and runs successfully.