National University of Computer and Emerging Sciences



# Lab Manual

*for*

# Data Structures

| Course Instructor | Ma’am Arooj Khalil |
| --- | --- |
| Lab Instructor(s) | Mr. Dilawar Shabbir  Mr. Sohaib Ahmad |
| Section | BSE-3A |
| Semester | FALL 2022 |

Department of Computer

Science FAST-NU, Lahore,

Pakistan

## Lab Manual 06

**Objectives:**

After performing this lab, students shall be able to revise:

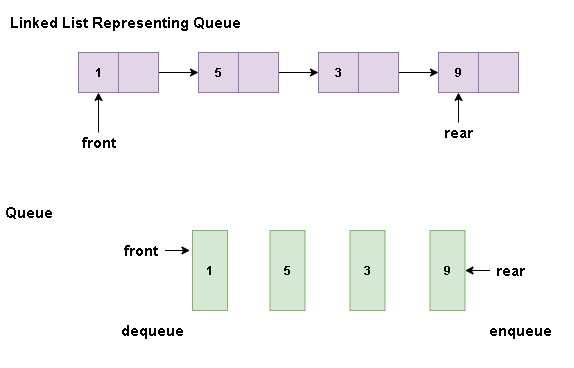
✔ Implementation of a Queue ADT using Linked List, Array and stack

NOTE: Create a separate file for each task.

**Q1. Implement a class Queue Using Link List**

Implement the following functions

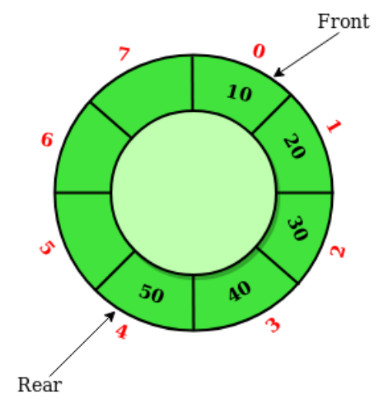
1. **IsFull:**
2. **IsEmpty:**
3. **Enqueue:** Add an element to the queue.
4. **Dequeue:** Removes the front element from the queue.
5. **Print:** It will print all elements of the queue in FIFO order

****

**Q2. Implement a class Queue Using Array as a circular list**

Implement the following functions

1. **IsFull:**
2. **IsEmpty:**
3. **Enqueue:** Add an element to the queue.
4. **Dequeue:** Removes the front element from the queue.
5. **Print:** It will print all elements of the queue in FIFO order

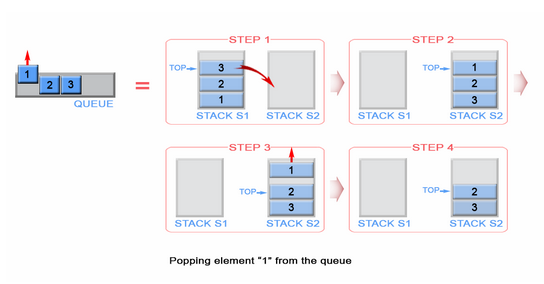


**Q3. You have to implement a template class queue using stack objects. The private members of the Queue class are just stack objects and you can only use operations of stacks.**

Implement the following functions:

1. **IsFull:**
2. **IsEmpty:**
3. **Enqueue:** Add an element in the queue.
4. **Dequeue:** Removes a front element from the queue.
5. **Print:** It will print all elements of the queue in FIFO order

NOTE: you are not allowed to use any array or link list in the queue class.



PROTOTYPE

* Queue(int Capacity) // create a fixed Capacity & Set the default values as well (for Q2.Q3, and for Q1 no need to set the capacity as LL is growable)
* ~Queue() // delete the memory allocated
* bool Enqueue(T Value)// for Q1 return type will be void
* bool Dequeue(T& ReturnValue) // if Queue is not empty return true and save the dequeued value in the passed parameter
* bool IsEmpty()
* bool IsFull() // This function is for Q2 and Q3.

***// add all calls of the function in the following main function that completely tests your implemented queue.***

***int main()***

***{***

***int value;***

***Queue<int> queue;***

***queue.Enqueue(9);***

***queue.Enqueue(8);***

***queue.Enqueue(7);***

***queue.Dequeue(value);***

***queue.Enqueue(2);***

***queue.Dequeue(value);***

***queue.Enqueue(3);***

***queue.Dequeue(value);***

***queue.Enqueue(4);***

***queue.Dequeue(value);***

***queue.Enqueue(5);***

***queue.Dequeue(value);***

***queue.Dequeue(value);***

***queue.Enqueue(50);***

***queue.Enqueue(2);***

***queue.Enqueue(10);***

***queue.Display(); // function to display whole queue---1***

***return 0;***

***}***

***Good Luck!***