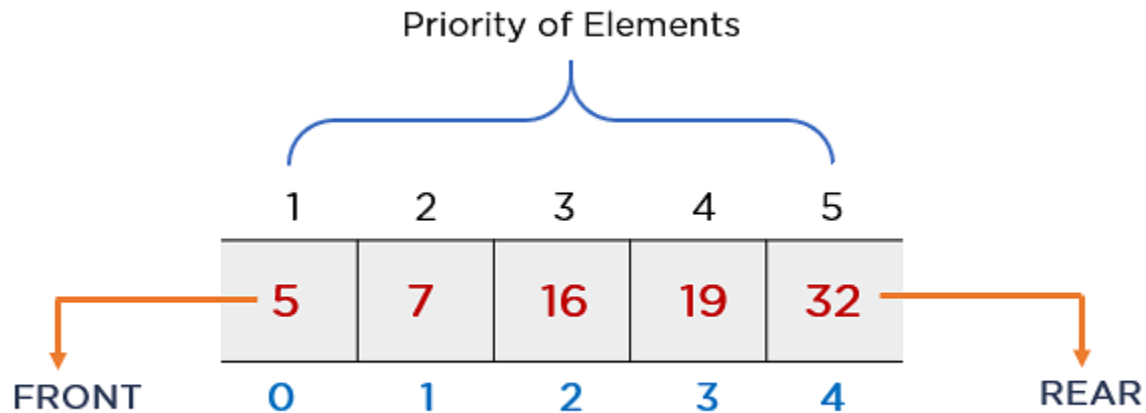


# Data Structures and Algorithms

## **Lecture 19:** Priority Queue and Dequeue

# Priority Queue

A priority queue is a type of queue that arranges elements based on their priority values. Elements with higher priority values are typically retrieved before elements with lower priority values.

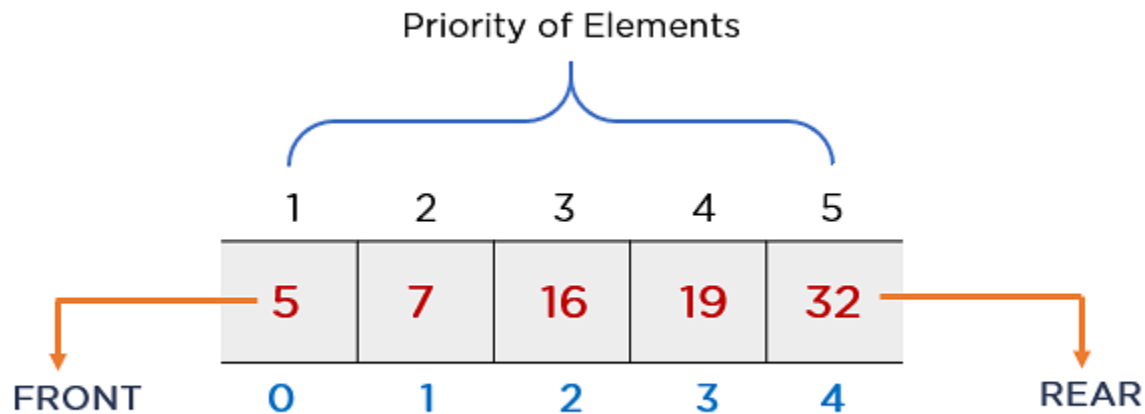


# Priority Queue

- 1) In a priority queue, each element has a priority value associated with it. When you add an element to the queue, it is inserted in a position based on its priority value. For example, if you add an element with a high priority value to a priority queue, it may be inserted near the front of the queue, while an element with a low priority value may be inserted near the back.
- 2) There are several ways to implement a priority queue, including using an array, linked list, heap, or binary search tree. Each method has its own advantages and disadvantages, and the best choice will depend on the specific needs of your application.

# Priority Queue

The priority queue in data structure resembles the properties of the hospital emergency queue. Thus, it is highly used in sorting algorithms. It behaves similar to a linear queue except for the fact that each element has some priority assigned to it. The priority of elements determines the order of removal in a queue, i.e., the element with higher priority will leave the queue first, whereas the element with the lowest priority at last.



# Properties of Priority Queue

- Every item has a priority associated with it.
- An element with high priority is dequeued before an element with low priority.
- If two elements have the same priority, they are served according to their order in the queue.

# Difference between Priority Queue and Normal Queue

- There is no priority attached to elements in a queue, the rule of first-in-first-out(FIFO) is implemented whereas, in a priority queue, the elements have a priority. The elements with higher priority are served first.

# Priority Queue Implementation

Priority queue can be implemented using the following data structures:

- Arrays
- Linked list
- Heap data structure
- Binary search tree

# Applications of Priority Queue

- CPU Scheduling
- Graph algorithms like Dijkstra's shortest path algorithm, Prim's Minimum Spanning Tree, etc.
- Stack Implementation
- All queue applications where priority is involved.
- Data compression in Huffman code
- Event-driven simulation such as customers waiting in a queue.
- Finding Kth largest/smallest element.



# Double-ended queue

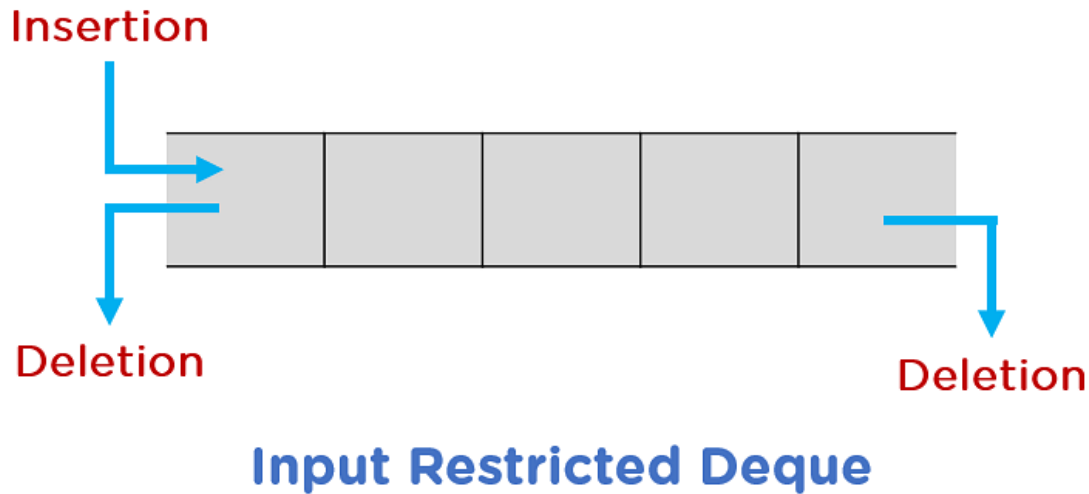
- A double-ended queue (dequeue or deque) is an abstract data type that generalizes a queue, for which elements can be added to or removed from either the front or rear.



# Double-ended queue

This general data class has some possible sub-types:

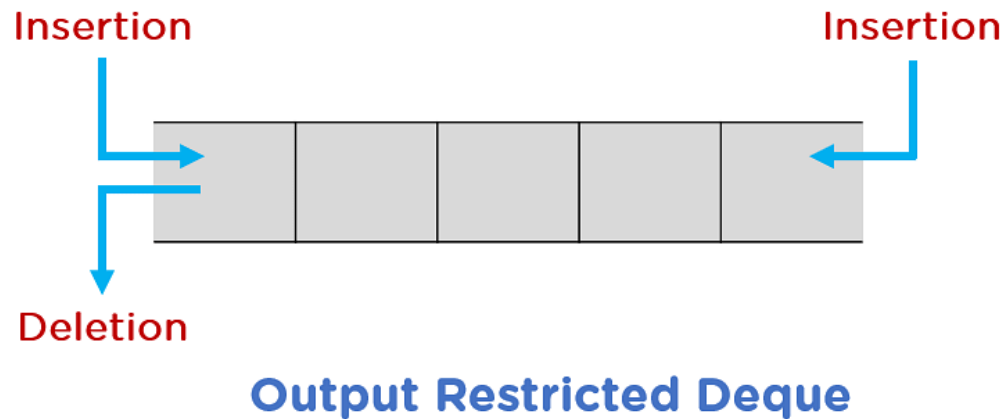
- 1) An input-restricted deque is one where deletion can be made from both ends, but insertion can be made at one end only.



# Double-ended queue

This general data class has some possible sub-types:

2) Output-Restricted Deque: It is a deque with some limitations while performing deletion operations



# Operations on Deque

Four basic operations are performed on deque, they are as follows:

Insertion at Rear

Insertion at Front

Deletion at Front

Deletion at Rear

# Recall:

- Priority Queue
- Deque

