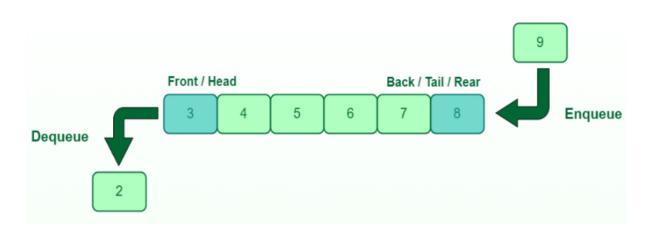# Queue

A queue is a linear data structure that follows the First-In-First-Out (FIFO) principle. It operates like a line where elements are added at one end (rear) and removed from the other end (front).

Queue Data Structure

The time complexity of a queue is O(1) for the enqueue and dequeue operations. This is because these operations only involve adding or removing an element from the front or back of the queue, respectively.

However, it is important to note that the time complexity of accessing or editing an element in a queue is O(n).

Basic Operations of Queue Data Structure

+ Enqueue (Insert): Adds an element to the rear of the queue.

+

Dequeue (Delete): Removes and returns the element from the front of the queue.
Peek: Returns the element at the front of the queue without removing it.
Empty: Checks if the queue is empty.
Full: Checks if the queue is full.

Applications of Queue

- Task scheduling in operating systems
- Data transfer in network communication

Queue

- Simulation of real-world systems (e.g., waiting lines)
- Priority queues for event processing queues for event processing