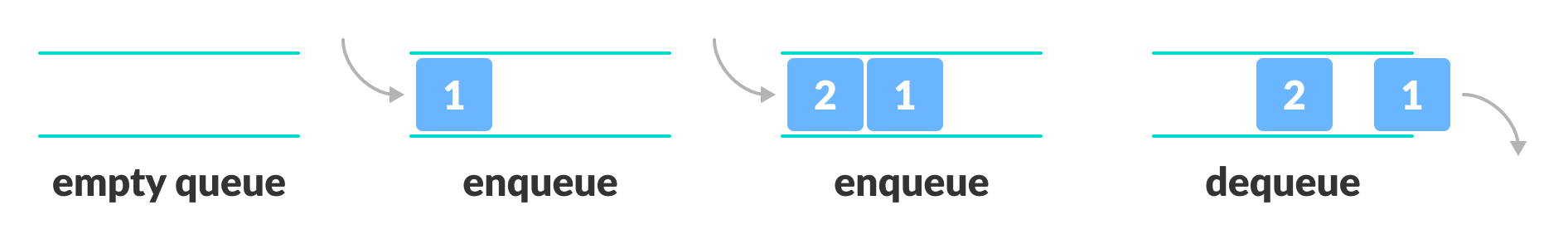
**Queue Data Structure**

A queue is a useful data structure in programming. It is similar to the ticket queue outside a cinema hall, where the first person entering the queue is the first person who gets the ticket.

Queue follows the **First In First Out (FIFO)** rule - the item that goes in first is the item that comes out first.



## Basic Operations of Queue

A queue is an object (an abstract data structure - ADT) that allows the following operations:

* **Enqueue**: Add an element to the end of the queue
* **Dequeue**: Remove an element from the front of the queue
* **Is Empty**: Check if the queue is empty
* **Is Full**: Check if the queue is full
* **Peek**: Get the value of the front of the queue without removing it
* **Size**: number of elements in the queue
* **Traverse:** loop through all elements in the queue
* **Clear :** remove all elements from queue

## Working of Queue

Queue operations work as follows:

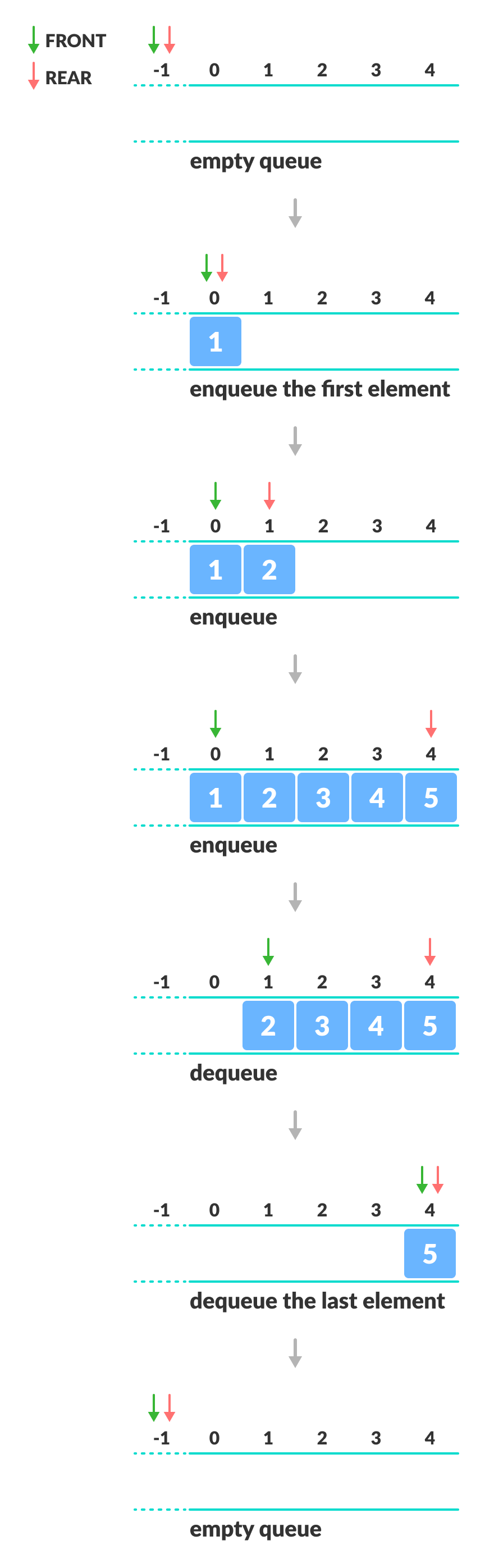
* two pointers FRONT and REAR
* FRONT track the first element of the queue
* REAR track the last element of the queue
* initially, set value of FRONT to 0 and REAR to -1

### Enqueue Operation

* check if the queue is full
* increase the REAR index by 1 (with % length of queue )
* add the new element in the position pointed to by REAR

### Dequeue Operation

* check if the queue is empty
* return the value pointed by FRONT
* increase the FRONT index by 1
* for the last element, reset the values of FRONT and REAR to -1

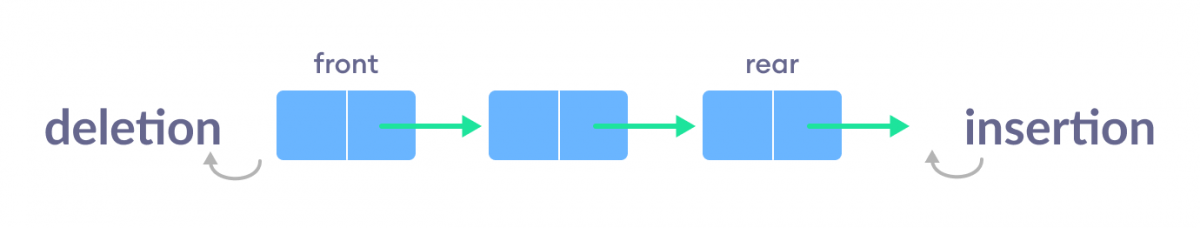


There are four different types of queues:

* Simple Queue
* Circular Queue
* Priority Queue
* Double Ended Queue

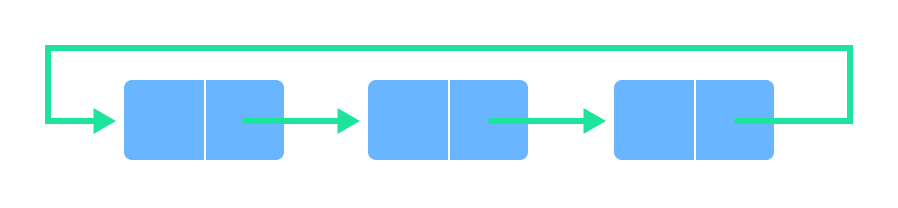
## Simple Queue

In a simple queue, insertion takes place at the rear and removal occurs at the front. It strictly follows the FIFO (First in First out) rule.



## Circular Queue

In a circular queue, the last element points to the first element making a circular link.



The main advantage of a circular queue over a simple queue is better memory utilization. If the last position is full and the first position is empty, we can insert an element in the first position. This action is not possible in a simple queue.

Chart, box and whisker chart

Description automatically generated

Chart, box and whisker chart

Description automatically generated

## Priority Queue

A priority queue is a **special type of queue** in which each element is associated with a **priority value**. And, elements are served on the basis of their priority. That is, higher priority elements are served first.

However, if elements with the same priority occur, they are served according to their order in the queue.

.

### Difference between Priority Queue and Normal Queue

In a queue, the **first-in-first-out rule** is implemented whereas, in a priority queue, the values are removed **on the basis of priority**. The element with the highest priority is removed first.

Text, table

Description automatically generated

Table

Description automatically generated

### Queue with linked list

