

# Queue Data Structure

# Examples of Queue

- People moving on an **escalator**. The people who got on the escalator first will be the first one to step out of it.
- People waiting for a **bus**. The first person standing in the line will be the first one to get into the bus.
- People standing outside the **ticketing window of a cinema hall**. The first person in the line will get the ticket first and thus will be the first one to move out of it.
- Luggage kept on **conveyor belts**. The bag which was placed first will be the first to come out at the other end.
- Cars lined at a **toll bridge**. The first car to reach the bridge will be the first to leave.

# Queue

- A queue is a FIFO (First-In, First-Out) data structure in which the element that is inserted first is the first one to be taken out.
- The elements in a queue are added at one end called the **REAR** and removed from the other end called the **FRONT**.
- Queues can be implemented by using either arrays or linked lists.

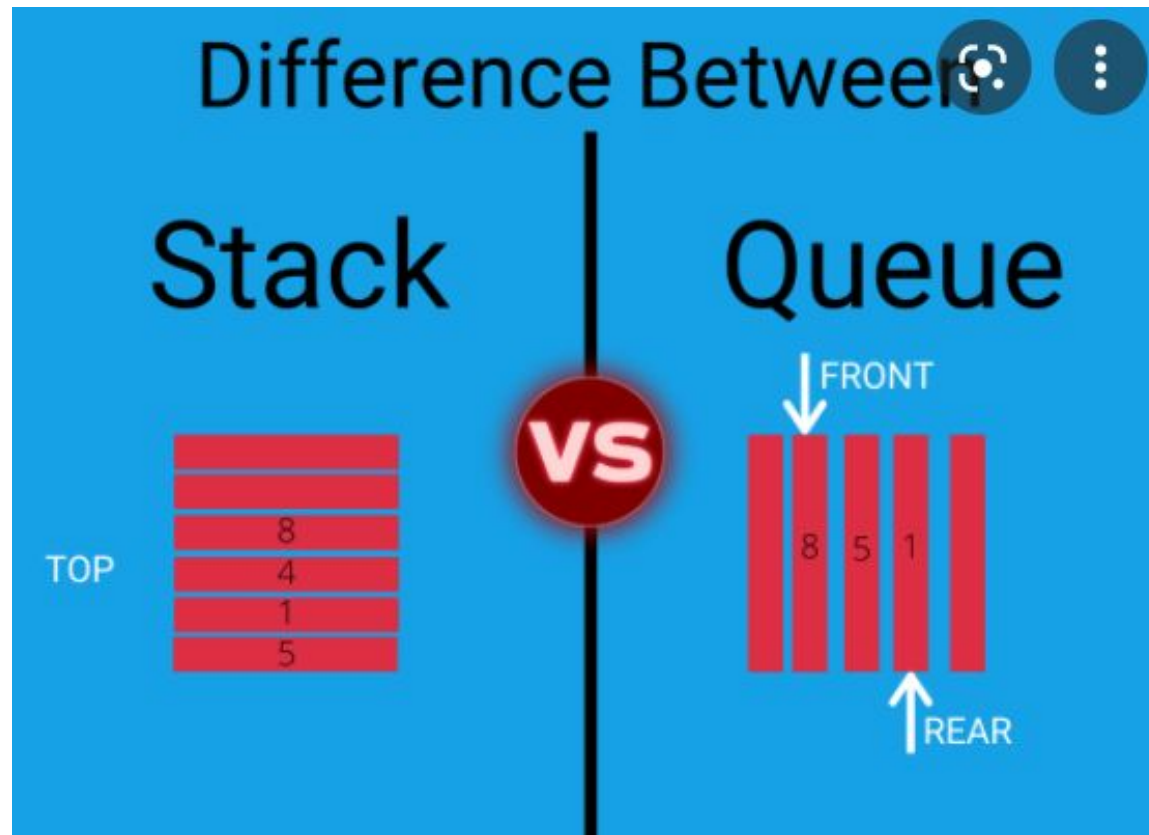
# Queue



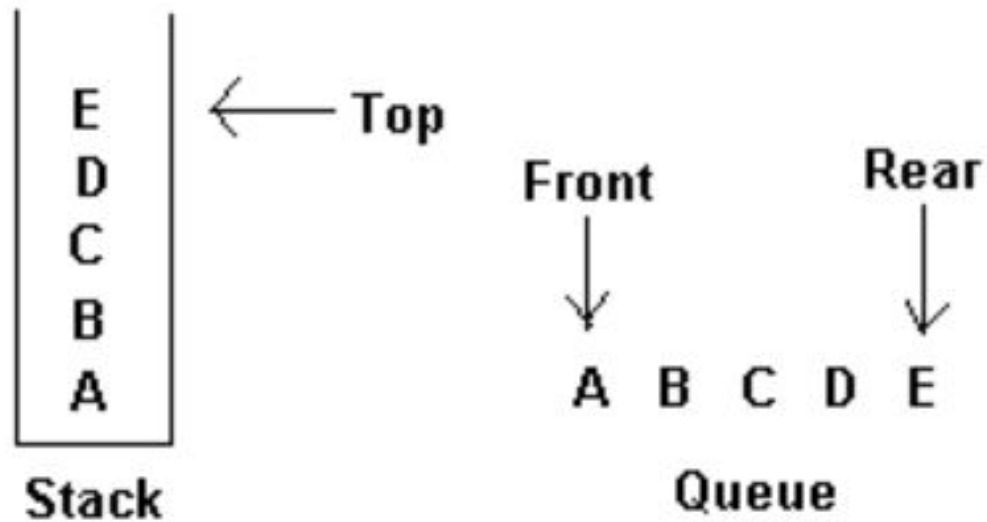
# Applications of Queue

- Queues are widely used as waiting lists for a single shared resource like printer, disk, CPU.
- Queues are used in asynchronous transfer of data (where data is not being transferred at the same rate between two processes) for eg. pipes, file IO, sockets.
- Queues are used as buffers in most of the applications like MP3 media player, CD player, etc.
- Queue are used to maintain the play list in media players in order to add and remove the songs from the play-list.
- Queues are used in operating systems for handling interrupts.

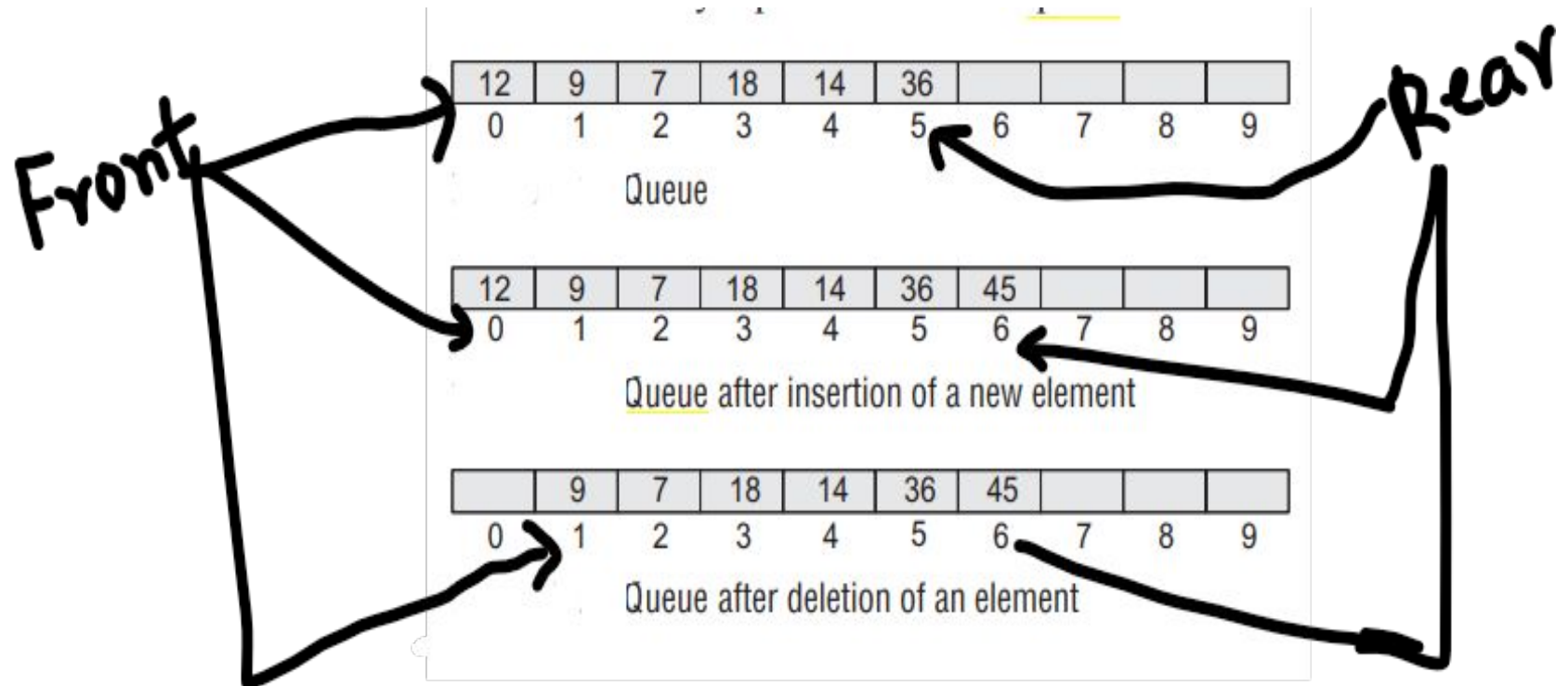
# Different between Stack and Queue



# Stack and Queue



# Array representation of Queue





# Array implementation of Queue

```
Step 1: IF REAR = MAX-1
        Write OVERFLOW
        Goto step 4
    [END OF IF]
Step 2: IF FRONT = -1 and REAR = -1
        SET FRONT = REAR = 0
    ELSE
        SET REAR = REAR + 1
    [END OF IF]
Step 3: SET QUEUE[REAR] = NUM
Step 4: EXIT
```

Algorithm to insert an element in  
a queue

```
Step 1: IF FRONT = -1 OR FRONT > REAR
        Write UNDERFLOW
    ELSE
        SET VAL = QUEUE[FRONT]
        SET FRONT = FRONT + 1
    [END OF IF]
Step 2: EXIT
```

Algorithm to delete an element from  
a queue

# Types of Queues

A queue data structure can be classified into the following types:

1. Circular Queue
2. Deques
3. Priority Queue
4. Multiple Queue