DSA Lab
Lab Experiment 5

Name: Aamir Ansari **Roll no:** 01 **Batch:** A

Aim: Implementation of circular queue using array.

Theory:

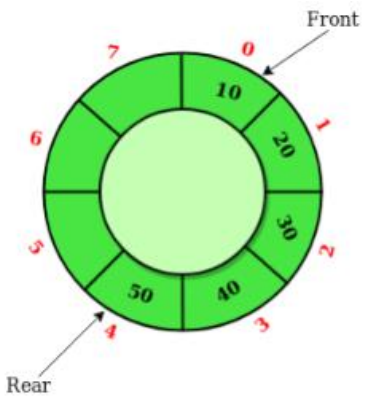
Drawbacks of linear queue:

We will explain the concept of circular queues using an example.

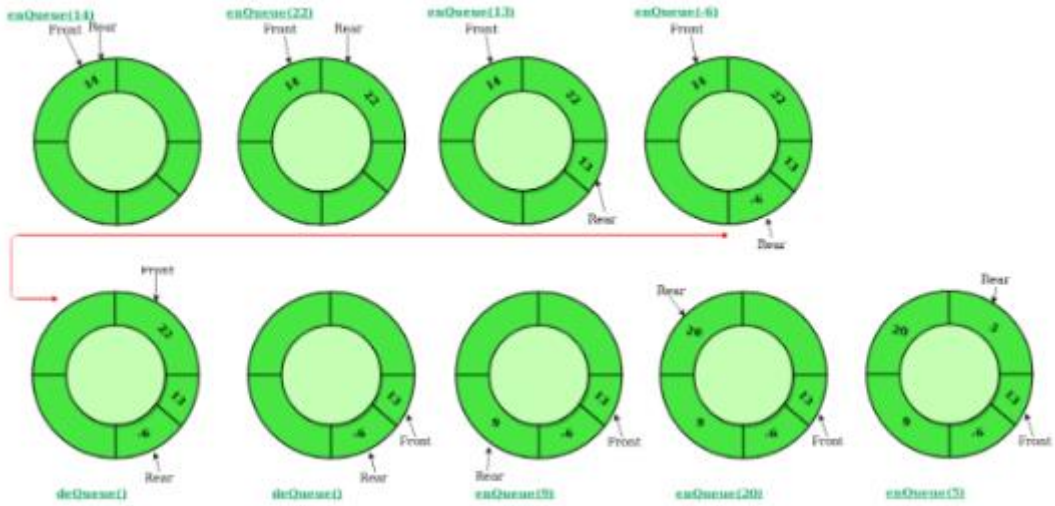
1. In this queue, front = 2 and rear = 9.
2. Now, if you want to insert a new element, it cannot be done because the space is available only at the left of the queue.
3. If rear = MAX – 1, then OVERFLOW condition exists.
4. This is the major drawback of a linear queue. Even if space is available, no insertions can be done once rear is equal to MAX – 1.
5. This leads to wastage of space. In order to overcome this problem, we use circular queues.
6. In a circular queue, the first index comes right after the last index.
7. A circular queue is full, only when front=0 and rear = Max – 1.

Circular queue:

Circular Queue is a linear data structure in which the operations are performed based on FIFO (First In First Out) principle and the last position is connected back to the first position to make a circle. It is also called ‘Ring Buffer’.



In a normal Queue, we can insert elements until queue becomes full. But once queue becomes full, we can not insert the next element even if there is a space in front of queue



Algorithms:

Algorithm to Insert an Element in a Circular Queue

Step 1: IF (FRONT = 0 and Rear = MAX – 1) OR (FRONT=Rear+1) Then

Write “OVERFLOW”

Goto Step 4

[END OF IF]

Step 2: IF FRONT = -1 and REAR = -1, then;

SET FRONT = REAR = 0

ELSE IF REAR = MAX – 1 and FRONT != 0

SET REAR = 0

ELSE

SET REAR = REAR + 1

[END OF IF]

Step 3: SET QUEUE[REAR] = VAL

Step 4: Exit

Algorithm to Delete an Element from a Circular Queue

Step 1: IF FRONT = -1, then Write "Underflow"

Goto Step 4

[END OF IF]

Step 2: SET VAL = QUEUE[FRONT]

Step 3: IF FRONT = REAR

SET FRONT = REAR = -1

ELSE

IF FRONT = MAX -1

SET FRONT = 0

ELSE

SET FRONT = FRONT + 1

[END OF IF]

[END OF IF]

Step 4: EXIT