# Title: -

**Experiment No-12**

C++ program simulate the pizza order system using circular queue using array

# Objectives: -

Understand the concept of queue as a circular queue using array

.

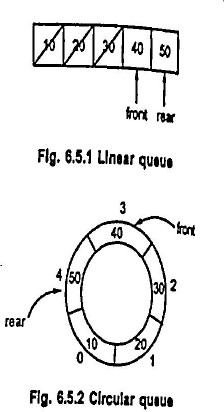
# Problem Statement: -

Pizza parlor accepting maximum M orders. Orders are served in first come first served basis. Order once placed cannot be cancelled. Write C++ program to simulate the system using circular queue using array

# Outcomes:-

The result of simulating the system using circular queue using array

# Theory-

**Basic Concept of circular queue:-**

As we have seen, in the case of linear queue the elements get deleted logically. This can be shown by following Fig. 65.1.

We have deleted the elements 10, 20 and 30 means simply the front pointer is shifted ahead. We will consider a queue from front to rear always. And now if we try to insert any more element then it won't be possible as it is going to give "queue full in message.

Although there is a space of elements 10,20 and 30 (these are deleted elements), we can not utilize them because queue is nothing but a linear array !

Hence there is a concept called circular queue. The main advantage of circular queue is we can utilize the space of the queue fully. The circular queue is shown by following Fig. 6.5.2.

Considering that the elements deleted are 10, 20 and

30.

There is a formula which has to be applied for incrementing the front and rear points, for a circular queue.

rear = (rear +1) % size front = (front +1) % size

If size of queue is 5 and rear is currently pointing at position 4 then after incrementing rear further

rear = (rear +1) % size

=(4+1) %5

rear = 0.

# Algorithm:-

**1.Add function( to add pizza order ‘ x ‘ in the queue)**

i) Increment rear counter circularly by 1 i.e. rear = (rear +1) % size

ii)Insert order at rear position as data[rear]=x

**2.Serve function( to serve order from the front position of queue)**

i) Increment front counter circularly by 1 front= (front +1) % size

ii)Server order x =data[front]

**3.Display function( to display all pending orders from the queue)**

i) Initialize i=front

ii)display data[i]

iii)increment i as i=(i+1)% size

iv)repeat step i), ii) ,iii) till front==0&&rear==size-1 || front==rear+1

**4.queuefull function( to check if queue is full)**

i)if (front==0&&rear==size-1 || front==rear+1 )

return 1to indicate queue is full

ii) else

return 0 to indicate queue is not full

**4.queueempty function( to check if queue is empty)**

i)if front =rear=-1

return 1to indicate queue is empty

ii) else

return 0 to indicate queue is not empty

**Conclusion:**

By this way, we can learn the concept of queue as a circular queue using array.