

Dry-Run

Circular Queue:

```
int main () {
```

```
    q.enqueue(1);
```

// Enqueue function.

```
void enqueue (1) {
```

```
    if (isFull()) {
```

```
        cout << "Queue is full";
```

```
    }
```

```
    else
```

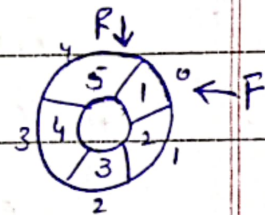
```
        if (front == -1)
```

```
            rear = rear + 1;
```

```
            arr[rear] = element;
```

→ Passing value.

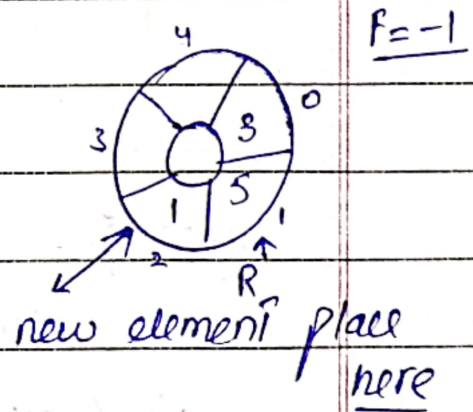
$F == 0$
 $r == \text{size} - 1$



```
        if (front == -1)
```

```
            rear = rear + 1;
```

```
            arr[rear] = element;
```



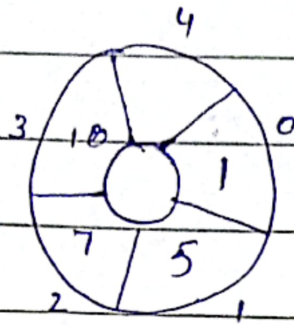
// let suppose

```
q.enqueue(5);
```

```
q.enqueue(7);
```

```
q.enqueue(10);
```

// The Queue.



q.dequeue();

// Dequeue function.

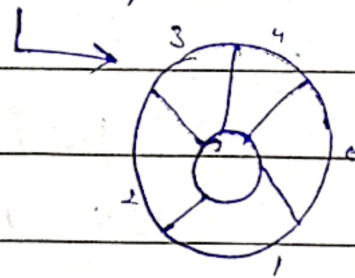
```
void dequeue() {
```

```
    int element;
```

```
    if (isEmpty()) {
```

```
        cout << "Queue is empty";
```

```
    }
```



F = -1
R = -1

```
else {
```

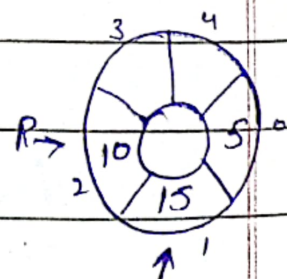
```
    element = item[front];
```

```
    if (front == rear) {
```

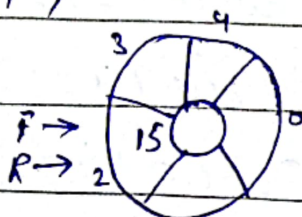
```
        front = -1;
```

```
        rear = -1;
```

```
    }
```



F = element
= 15



After deleting
last element both F, R equals to -1.

DATE: _____

DAY: _____

else {

$$\text{Front} = \text{Front} + 1$$

}

