Circular Queue

```
#define N 5
int front = -1, rear = -1;
void enqueue(int x){
if (front == -1 && rear == -1){
    front = rear = 0;
    queue[rear] = x;
 }else if (rear == N-1){ else if ((rear+1)%N == front){
else    rear = (rear+N)+1; printf("Overflow");}
  else { rear = (rear+1)%N
        queue[rear] = x;
   }}
  void dequeue(){
   if (front == -1){
     printf("Underflow");
   }else if (front == rear){
        front = rear = -1;}
  else { front = (front+1)%N
        queue[front] = item;
   printf("Deleted item is %d", item");
  }}
  void display(){
    if (front == -1){                  { if(i == rear){ break; }
      printf("Underflow");}              i = (i+1)%N;          }
  else { for(i=0; i<N; i++){ int i = front;
 while(1) { printf(queue[i]);
        printf("There are the elements of queue");
   }}
```

Output:

Enter your choice : 1. enqueue 2. dequeue 3. display 4. exit
1
Enter element : 2
Enter your choice : 1. enqueue 2. dequeue 3. display 4. exit
2
Deleted element : 2
Enter your choice : 1. enqueue 2. dequeue 3. display 4. exit
3
Queue is empty
Enter your choice : 1. enqueue 2. dequeue 3. display 4. exit.
4
Exiting

4/11/25