Lab - 03

Q) WAP to stimulate the working of a queue of integers using an array. Provide the following operations: Insert, Delete, Display The Program should print appropriate msgs for queue.

Pseudo Code

Starting the program

Setting the front & rear of array = -1

enque {

```
if (rear = N-1) {
    printf("queue overflow\n")
}
else if (front = rear = N-1) {
    front = rear = 0;
    queue[rear] = x;        // x refers to the integers to be queued
}
}
else {
    queue[rear] = x;
}
```

deque {

```
if ((front = rear = -1)) {
    printf("queue empty");
}
else if (front = rear) {
    front = rear = -1}
else {
    return queue[front++]
}
```

display () {

```
if front = rear = -1 {
    print queue empty
}
else { for int i = front ; i<=rear i++) {
    print [i]
}
}
```

peek () {

```
if (front = rear = -1 {
    printf("queue empty");
}
else {
    return queue[front]
}
```

```
#include <stdio.h>
#define N 2

int front = -1;
int rear = -1;
int queue[N];

void enque(int x)
{
    if (front == -1 && rear == -1) {
        front = rear = 0;
        queue[rear] = x;
    }
    else if (rear == N-1) {
        printf("Queue Overflow\n");
    }
    else {
        queue[++rear] = x;
    }
}

int deque(){
    if (front == -1 && rear == -1) {
        printf("Queue is empty\n");
        return -1;
    }
    else if (front == rear) {
        int element = queue[front];
        front = rear = -1;
        return element;
    }
    else {
        return queue[front++];
    }
}
void display()
{
    if (front == -1 && rear == -1) {
        printf("Queue Empty\n");
    }
    else {
        for (int i = front; i <= rear; i++) {
            printf("%d\n", queue[i]);
        }
    }
}

int peek() {
    if (front == -1 && rear == -1) {
        printf("Queue Empty\n");
        return -1;
    }
    else {
        return queue[front];
    }
}
void main() {
    int choice = 0;
    while (choice != 5) {
        printf("\n Enter 1 for enque, 2 for deque,
                3 for peek, 4 for display, 5 to exit:");
        scanf("%d", &choice);
```