```c
/*
Title- Queue Implimentation using array(statically)
Author- Bhakare Mahesh Santosh
ID- 492
Batch- TechnOrbit(PPA-8)
*/

#include<stdio.h>
#define MAX 10

// ----------------------- STRUCTURE DECLERATION --------------------------------

struct QUEUE
{
    int arr[MAX];
    int front;
    int rear;
};

// ------------------ FUNCTION TO INITILIZE QUEUE -------------------------

void init_queue(struct QUEUE* queue)
{
    queue->front = queue->rear = -1;
}


// ---------------------- FUNCTION TO CHECK QUEUE FULL OR NOT ------------------

int full(struct QUEUE* queue)
{
    if(queue->rear == MAX-1)
    {
        return 1;
    }
    else
    {
        return 0;
    }
}

// ----------------------- FUNCTION TO CHECK EMPTY QUEUE -------------------------

int empty(struct QUEUE* queue)
{
    if(queue->rear == queue->front)
    {
        queue->rear = queue->front = -1;
    }
    if(queue->rear == -1)
    {
        return 1;
    }
    else
    {
        return 0;
    }
}

// ------------------------------- FUNCTION TO INSERT ELEMENT IN QUEUE
--------------------

void enqueue(struct QUEUE* queue)
{
    (queue->rear)++;
```

```
    printf("Please Enter value: ");
    scanf("%d",&(queue->arr[queue->rear]));
}

// ------------------------------- FUNCTION TO DELETE ELEMENT FROM QUEUE
----------------------

void dequeue(struct QUEUE* queue)
{
    (queue->front)++;
    printf("Dequeued Element is: %d\n",queue->arr[queue->front]);
}

//  ------------------------------- FUNCTION TO DISPLAY QUEUE
-------------------------------

void display(struct QUEUE* queue)
{
    int i;
    printf("Your Queue is: ");
    for(i=(queue->front)+1;i<=queue->rear;i++)
    {
        printf("%d <=",queue->arr[i]);
    }
    printf("\n");
}

// ------------------------------------ ENTRY POINT FUNCTION -------------------------

void main()
{
    int choice;
    struct QUEUE queue;
    init_queue(&queue);
    do
    {
        printf("**********************************\n");
        printf("1. Enqueue\n2. Dequeue\n3. IsFull\n4. IsEmpty\n5. Display\n6. Exit\nEnter
Your Choice: ");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1: if(full(&queue))
                    {
                        printf("Queue is Full.........\n");
                    }
                    else
                    {
                        enqueue(&queue);
                    }
                    break;

            case 2: if(empty(&queue))
                    {
                        printf("Queue is Empty.....\n");
                    }
                    else
                    {
                        dequeue(&queue);
                    }
                    break;

            case 3: if(full(&queue))
                    {
                        printf("Queue is full....\n");
                    }
```

```
                    else
                    {
                        printf("%d positions vaccent....\n",(MAX-queue.rear-1));
                    }
                    break;
            case 4: if(empty(&queue))
                    {
                        printf("Queue is empty....\n");
                    }
                    else
                    {
                        printf("%d elements in queue...\n",(queue.rear - queue.front));
                    }
                    break;

            case 5: if(empty(&queue))
                    {
                        printf("Queue is Empty...\n");
                    }
                    else
                    {
                        display(&queue);
                    }
                    break;

            case 6: printf("Exiting.........\n");
                    break;

            default : printf("Wrong choice please enter proper choice.........\n");
        }

    }while(choice != 6);

}
```