

Institute of Computer Technology
B. Tech Computer Science and Engineering
Subject: DS (2CSE302)

PRACTICAL-12

AIM: - Implement the scenario based on circular Queue using array.

In a computer-control traffic system, traffic light is one of the best examples of the circular queue. Each light of traffic light gets ON one by one after every interval of time. Like red light gets ON for one minute then yellow light for one minute and then green light. After green light, the red light gets ON.

If there is a maximum capacity of the circular queue is 3 (Implement the below circular queue operation using C language:

- **insert (ENQUEUE)**
- **delete (DEQUEUE),**
- **display (TRACING)**

Input: Insert 10

Insert 30

Insert 20

Delete

Insert 40

Output:

Queue is empty !!

Inserted -> 10

Inserted -> 30

Inserted -> 20

Queue is full!!

Front -> 0

Items -> 10 30 20

Rear -> 2

Deleted element -> 10

Front -> 1

Items -> 30 20

Rear -> 2

Inserted -> 40

Front -> 1

Items -> 30 20 40

Rear -> 0

Queue is full!!

SOLUTION

```
#include <stdio.h>
#include <stdlib.h>
#define MAX 3

int front = - 1;
int rear = - 1;

void INSERT(int Yash[])
{
    int n;
    if ((front == 0 && rear == MAX - 1) || (front == rear + 1))
    {
        printf("\n.....Queue is full.....");
        return;
    }
    else if (rear == - 1)
    {
        rear++;
        front++;
    }
    else if (rear == MAX - 1 && front > 0)
    {
        rear = 0;
    }
    else
    {
        rear++;
    }
    printf("\nEnter number: ");
    scanf("%d", &n);

    Yash[rear] = n;
    printf("\nINSERTED -> %d",n);
}

void DELETE(int Yash[])
{
    if (front == - 1)
    {
        printf("\n.....Queue is empty.....");
    }
    else if (front == rear)
    {

```

```
printf("\nDeleted -> %d", Yash[front]);
front = - 1;
rear = - 1;
}
else
{
    printf("\nDeleted -> %d", Yash[front]);
    front++;
}
}

void DISPLAY(int Yash[])
{
    int i;
    if (front == -1 & rear == -1)
    {
        printf("\n.....Queue is Empty.....\n");
    }
    else
    {
        if (front > rear)
        {
            printf("\nITEMS -> ");
            for (i = front; i < MAX; i++)
            {
                printf("%d ", Yash[i]);
            }
            for (i = 0; i <= rear; i++)
            {
                printf("%d ", Yash[i]);
            }
            printf("\nFRONT -> %d", front);
            printf("\nREAR -> %d", rear);
        }
        else
        {
            printf("\nITEMS ->");
            for (i = front; i <= rear; i++)
            {
                printf("%d ", Yash[i]);
            }
            printf("\nFRONT -> %d", front);
            printf("\nREAR -> %d", rear);
        }
    }
}
```

```
    }  
}  
  
int main()  
{  
    int ch;  
    int Yash[MAX];  
    while(1)  
    {  
        RETRY:  
        printf("\n-----| CIRCULAR QUEUE |-----");  
        printf("\n1. Insert \n2. Delete\n3. Display\n4. Exit");  
        printf("\nEnter Your Choice: ");  
        scanf("%d", &ch);  
        switch (ch)  
        {  
            case 1:  
                INSERT(Yash);  
                break;  
            case 2:  
                DELETE(Yash);  
                break;  
            case 3:  
                DISPLAY(Yash);  
                break;  
            case 4:  
                printf("Exiting system.....\n\n");  
                exit(0);  
                break;  
            default:  
                printf("\n.....INVALID CHOICE !!.....\n.....Enter Again.....");  
                goto RETRY;  
                break;  
        }  
    }  
    return 0;  
}
```

OUTPUT

```
CentOS 8 64-bit - VMware Workstation
File Edit View VM Tabs Help
Library
CentOS 8 64-bit
My Computer
CentOS 8 64-bit
Shared VMs (Deprecated)
Activities Applications Terminal
Oct 8 01:47
yash@localhost:~/Desktop/DS/practicals/Prac12
File Edit View Search Terminal Help
[yash@localhost Prac12]$ gedit p12.c
[yash@localhost Prac12]$ gcc p12.c
[yash@localhost Prac12]$ ./a.out

----| CIRCULAR QUEUE |----
1. Insert
2. Delete
3. Display
4. Exit
Enter Your Choice: 3

....Queue is Empty....

----| CIRCULAR QUEUE |----
1. Insert
2. Delete
3. Display
4. Exit
Enter Your Choice: 1

Enter number: 10

INSERTED -> 10
----| CIRCULAR QUEUE |----
1. Insert
2. Delete
3. Display
4. Exit
Enter Your Choice: 1

Enter number: 30
```

```
CentOS 8 64-bit - VMware Workstation
File Edit View VM Tabs Help
Library
CentOS 8 64-bit
My Computer
CentOS 8 64-bit
Shared VMs (Deprecated)
Activities Applications Terminal
Oct 8 01:47
yash@localhost:~/Desktop/DS/practicals/Prac12
File Edit View Search Terminal Help
Enter number: 30

INSERTED -> 30
----| CIRCULAR QUEUE |----
1. Insert
2. Delete
3. Display
4. Exit
Enter Your Choice: 1

Enter number: 20

INSERTED -> 20
----| CIRCULAR QUEUE |----
1. Insert
2. Delete
3. Display
4. Exit
Enter Your Choice: 3

ITEMS ->10 30 20
FRONT -> 0
REAR -> 2
----| CIRCULAR QUEUE |----
1. Insert
2. Delete
3. Display
4. Exit
Enter Your Choice: 1

Queue is full
```

