EXPERIMENT-5

Program :

```
#include <stdio.h>
int FRONT=0 ,REAR = -1, size=0, capacity;
void EnQ(int CircularQ[],int value){
    if(size==capacity){return;}
    else{
        REAR = (FRONT + size)%capacity;
        CircularQ[REAR]=value;
        size++;
    }
}
void DeQ(){
    if(size==0){return;}
    if(FRONT==REAR){
        FRONT=0;
        REAR =-1;
    }
    else{
        FRONT = (FRONT+1)%capacity;
    }
    size--;
}
void display(int CircularQ[]){
    printf("Circular Queue: ");
    if(FRONT<=REAR){
```

```
        for(int i=FRONT;i<=REAR;i++){
            printf("%d\t",CircularQ[i]);
        }
    }
    else{
        for(int i=FRONT;i<=capacity-1;i++){
            printf("%d\t",CircularQ[i]);
        }
        for(int i=0;i<=REAR;i++){
            printf("%d\t",CircularQ[i]);
        }
    }
}
int main(){
    printf("\nEnter the size of Circular Queue: ");
    scanf("%d",&capacity);
    int CircularQ[capacity];
while(1){
        int choice,value;
        printf("\nSelect queue option!\n");
        printf("1. Enqueue\t2. Dequeue\t3. Exit\nChoice: ");
        scanf("%d", &choice);

        switch(choice){
            case 1:if(size==capacity){
                printf("Queue Overflow!\n");
            }else{
```

```
            printf("\nEnter the element: \n");
            scanf("%d",&value);
            EnQ(CircularQ,value);
            display(CircularQ);
        }
        break;
        case 2:if(size==0){
            printf("Queue Empty!\n");
        }else{
            DeQ();
            display(CircularQ);
        }
        break;
        case 3: return 0;
        default:printf("Invalid choice!");
    }
  }
}
```

Output :

```
cseb2@sjcet-OptiPlex-SFF-7020:~/Alwin$ gcc circularQueue.c
cseb2@sjcet-OptiPlex-SFF-7020:~/Alwin$ ./a.out

Enter the size of Circular Queue: 2

Select queue option!
1. Enqueue      2. Dequeue      3. Exit
Choice: 1

Enter the element:
10
Circular Queue: 10
Select queue option!
1. Enqueue      2. Dequeue      3. Exit
Choice: 1

Enter the element:
12
Circular Queue: 10      12
Select queue option!
1. Enqueue      2. Dequeue      3. Exit
Choice: 1
Queue Overflow!

Select queue option!
1. Enqueue      2. Dequeue      3. Exit
Choice: 2
Circular Queue: 12
Select queue option!
1. Enqueue      2. Dequeue      3. Exit
Choice: 3
cseb2@sjcet-OptiPlex-SFF-7020:~/Alwin$
```