```c
#include <stdio.h>
#include <stdlib.h>
#define S 3

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

int queue[S];

void Enque(int, int);
int Deque(int);
void display(int);
int main(int argc, char **argv)
{
int choice, SIZE;
int item;
printf("Enter the SIZE of the queue : \n");
scanf("%d",&SIZE);
do
{
printf("\n<---------CIRCULAR QUEUE--------->\n");
printf("\n 1. INSERT to Queue (EnQueue)");
printf("\n 2. DELETE from the Queue (DeQueue)");
printf("\n 3. DISPLAY the content ");
printf("\n 4. EXIT\n");
printf("\n<------------------------------->\n");
printf("Enter your choice accordingly : ");
scanf("%d", &choice);
switch(choice)
{
```

```c
        case 1:   if(((front == 0 && rear == SIZE -
        1)) || (front == rear + 1))
        {
        printf("-----Queue is FULL-----\n");
        break;
        }

        printf("\nEnter the element you want to
        INSERT : \n");
        scanf("%d",&item);
        Enque(SIZE, item);
        break;

        case 2: item=Deque(SIZE);
        if(item==-999)
        printf("-----Queue is EMPTY-----\n");
        else
        printf("\nRemoved element from the queue
        %d\n",item);
        break;

        case 3: display(SIZE);
        break;

        case 4: printf("EXITING......\n");
        exit(0);
        default: printf("INVALID CHOICE!!");
        break;
        }
        }
        while (choice!=4);
        return 0;
        }

        void Enque(int SIZE, int ele)
        {
```

```c
        if(((front == 0 && rear == SIZE - 1)) ||
        (front == rear + 1) )
        {
        printf("-----Queue is FULL-----\n");
        return;
        }
        else
        {
        rear=(rear+1)%SIZE;
        queue[rear]=ele;
        if(front == -1)
        front=0;
        }
        }

        int Deque(int SIZE)
        {
        int item;
        if((front == -1)&&(rear == -1))
        {
        return(-999);
        }
        else
        {
        item=queue[front];

        if(front==rear)
        {
        front=-1;
        rear=-1;
        }
        else
        {
        front=(front+1)%SIZE;
        }
        return item;
```

```c
                              }


                              }


                              void display(int SIZE)
                              {
                              int i;
                              if(((front==-1)&& (rear==-1)))
                              {
                              printf("-----Queue is EMPTY-----\n");
                              return;
                              }
                              else
                              {
                              printf("\nQueue contents:\n");
                              for(i=front;i!=rear;i=(i+1)%SIZE)
                              {
                              printf("%d\t", queue[i]);
                              }
                              printf("%d\t", queue[i]);
                              }
                              }
```

```
Enter the SIZE of the queue :
4

<----------CIRCULAR QUEUE---------->

 1. INSERT to Queue (EnQueue)
 2. DELETE from the Queue (DeQueue)
 3. DISPLAY the content
 4. EXIT

<--------------------------------->
Enter your choice accordingly : 1

Enter the element you want to INSERT :
12

<----------CIRCULAR QUEUE---------->

 1. INSERT to Queue (EnQueue)
 2. DELETE from the Queue (DeQueue)
 3. DISPLAY the content
 4. EXIT

<--------------------------------->
Enter your choice accordingly : 1
```

```
Enter the element you want to INSERT :
23


<---------CIRCULAR QUEUE--------->


 1. INSERT to Queue (EnQueue)
 2. DELETE from the Queue (DeQueue)
 3. DISPLAY the content
 4. EXIT


<------------------------------->
Enter your choice accordingly : 3


Queue contents:
12      23
<---------CIRCULAR QUEUE--------->


 1. INSERT to Queue (EnQueue)
 2. DELETE from the Queue (DeQueue)
 3. DISPLAY the content
 4. EXIT


<------------------------------->
Enter your choice accordingly : 2
```

```
Removed element from the queue 12


<---------CIRCULAR QUEUE--------->


 1. INSERT to Queue (EnQueue)
 2. DELETE from the Queue (DeQueue)
 3. DISPLAY the content
 4. EXIT


<------------------------------->
Enter your choice accordingly : 3


Queue contents:
23
<---------CIRCULAR QUEUE--------->


 1. INSERT to Queue (EnQueue)
 2. DELETE from the Queue (DeQueue)
 3. DISPLAY the content
 4. EXIT


<------------------------------->
```