circular queue.c

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

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

int queue[MAX];

void Enque(int);
int Deque();
void display();
int main(int argc, char **argv)
{
    int choice;
    int item;
    do{

        printf("\n 1. Insert to Queue ");
         printf("\n 2. delete from the Queue ");
        printf("\n 3. Display the content ");
        printf("\n 4. Exit\n");
        printf("Enter your Choice\n :");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1:  printf("Enter the element to be enqueued:")
                     scanf("%d",&item);
                     Enque(item);
                     break;
            case 2:  item=Deque();
```

Scanned by CamScanner

circular queue.c

```
            case 1:  printf("Enter the element to be enqueued:");
                     scanf("%d",&item);
                     Enque(item);
                     break;
            case 2: item=Deque();
                    if(item==-1)
                        printf("Queue is empty");
                    else
                    printf("Dequeued element from the queue %d",item);
                    break;
                case 3: display();
                    break;
                case 4: exit(0);
                default :printf("enterd wrong choice\n");
        }
    } while (choice!=4);
    return 0;
}

void Enque(int ele)
{
    if(((front == 0 && rear == MAX - 1))|| (front == rear + 1) )
    {
       printf("Queue is full\n");
       return;

    }
    else
    {
    rear=(rear+1)%MAX;
    queue[rear]=ele;
```

Scanned by CamScanner

Start here X | circular queue.c X

```
47  {
48      if(((front == 0 && rear == MAX - 1))|| (front == rear + 1) )
49      {
50          printf("Queue is full\n");
51          return;
52
53      }
54      else
55      {
56          rear=(rear+1)%MAX;
57          queue[rear]=ele;
58          if(front ==-1)
59              front=0;
60
61
62      }
63  }
64  int Deque()
65  {
66      int item;
67      if((front == -1)&&(rear == -1))
68      {
69
70          return(-1);
71      }
72      else
73      {
74          item=queue[front];
75
76          if(front==rear)
77          {
```

Scanned by CamScanner

```
76          if(front==rear)
77          {
78              front=-1;
79              rear=-1;
80          }
81          else
82          {
83              front=(front+1)%MAX;
84          }
85          return item;
86      }
87
88  }
89
90  void display()
91  {
92      int i;
93      if(((front==-1)&& (rear==-1))|| (front==rear))
94      {
95
96          printf("Queue is empty\n");
97          return;
98
99      }
100     else
101     {
102         printf("Elements in the Queue are:\n");
103         for(i=front;i<=rear;i++)
104             printf("%d\n", queue[i]);
105     }
```

Scanned by CamScanner