main.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 option;
    int item;
    do{
        printf("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("Enter the option :");
        scanf("%d",&option);
        switch(option)
        {
            case 1:  printf("Enter the element\n");
                     scanf("%d",&item);
                     Enque(item);
                     break;
            case 2: item=Deque();
                    if(item==-999)
```

```
                break;
            case 2: item=Deque();
                if(item==-999)
                    printf("Queue is empty");
                else
                printf("Removed element from the queue %d",item);
                break;
            case 3: display();
                break;
            case 4: exit(0);
        }
    } while (option!=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;
      if(front ==-1)
          front=0;


    }
 }
 int Deque()
```

```
62  int Deque()
63  {
64      int item;
65      if((front == -1)&&(rear == -1))
66      {
67
68          return(-1);
69      }
70      else
71      {
72          item=queue[front];
73
74          if(front==rear)
75          {
76              front=-1;
77              rear=-1;
78          }
79          else
80          {
81              front=(front+1)%MAX;
82          }
83          return item;
84      }
85
86  }
87
88  void display()
89  {
90    if (front == -1 && rear == -1)
91    {
92      printf("Queue is empty");
93    }
```

main.c

```
88  void display()
89  {
90      if (front == -1 && rear == -1)
91      {
92          printf("Queue is empty");
93      }
94      else
95      {
96          printf("Queue contents : \n");
97          if (front <= rear)
98          {
99              for (int i = front; i <= rear; i++)
100             {
101                 printf("%d\n", queue[i]);
102             }
103         }
104         else
105         {
106             for (int i = front; i <= MAX-1; i++)
107             {
108                 printf("%d\n", queue[i]);
109             }
110             for (int i = 0; i <= rear; i++)
111             {
112                 printf("%d\n", queue[i]);
113             }
114         }
115     }
116 }
117
```