

```

#include <stdio.h>

#include <stdlib.h>


#define max 5

typedef struct
{
    int rear,front;
    int arr[max];
}QUEUE;

void enqueue(QUEUE *q,int data)
{

    if((q->rear+1)%max==q->front)
    {
        printf("Queue is over flow\n");
        return;
    }
    if(q->front==-1)
        q->front=0;
    q->rear=(q->rear+1)%max;
    q->arr[q->rear]=data;
}

int dequeue(QUEUE *q)
{
    int data;

    data=q->arr[q->front];
    if(q->front==-1)
    {
        printf("Queue is empty\n");
        return -999 ;
    }
}

```

```

    }
    if(q->front==q->rear){
        q->rear=-1;
        q->front=-1;
        return data;
    }
    else
        q->front=(q->front+1)%max;

    return (data);
}

void display(Queue *q)
{
    int i;
    if(q->front== -1)
    {
        printf("Queue is empty\n");
        return ;
    }
    printf("the elements in circular queue is\n\n");
    for(i=q->front;;i=(i+1)%max)
    {
        printf("%d ",q->arr[i]);

        if(i==q->rear)
            break;
    }
    printf("\n");
}

int main()
{

```

```

    QUEUE q;

    int ch,data;

q.front=-1;
q.rear=-1;
    while(ch!=4)
    {

printf("\n\n1.Enqueue\t\t2.DeQueue\n3.Display\t\t4.Exit\n");
printf("enter your choice\n");
scanf("%d",&ch);
switch(ch)
{
    case 1:printf("Enter the data\n");
        scanf("%d",&data);
        enqueue(&q,data);
        break;
    case 2:data=dequeue(&q);
        if(data!=-999)
            printf("Deleted data is %d",data);
        break;
    case 3:display(&q);
        break;
    case 4:exit(0);
        break;
    default:printf("invalide input\n");
        break;

}

}

return 0;

```

