queue > C que.c > ...

```c
#include<stdio.h>
#define MAX 8
int f=-1;
int r=-1;
int queue[MAX];
void enqueue(int data)
{
    if(r==MAX-1)
    {
        printf("overflow");
    }
    else
    {
        if(f==-1)
        {
            f=0;
        }
        r++;
        queue[r]=data;

    }
}
```

queue > C que.c > ...

```c
}
void dequeue()
{
    if(f==-1 || f>r)
    {
        printf("underflow");
    }
    if (f == r) {

        f = r = -1;}
    else
    {
        printf("popped element is ");
        printf("%d",queue[f]);
        f++;
    }
}
void display()
{
    if(f==-1)
    {
        printf("empty");
    }
```

```c
void display()
    else
        int i;
        printf("elements in queue are\n");
        for(i=f;i<=r;i++)
        {
            printf("%d\n",queue[i]);
        }
    }
}
int peek()
{
    if(f==-1)
    {
        printf("empty");
        return -1;
    }
    else
    {
        return queue[f];
    }
}
int main()
```

```c
}
int main()
{
    enqueue(10);
    enqueue(20);
    enqueue(30);
    display();
    dequeue();
    printf("\nPeek: %d\n", peek());
    dequeue();
    display();


    printf("Peek after dequeue: %d\n", peek());
    return 0;

}
```

```
elements in queue are
10
20
30
popped element is 10
Peek: 20
popped element is 20elements in queue are
30
Peek after dequeue: 30
```

```
emelents are10 20 30 35 50
Peek: 10
popped element is 10 popped element is 20 Peek: 30
emelents are30 35 50 90
```

queue > C que.c > ...

```c
#include<stdio.h>
#define MAX 5
int f=-1;
int r=-1;
int queue[MAX];
void enqueue(int data)
{
    if((r+1)% MAX == f)
    {
        printf("overflow");
        return;
    }
    else{
        if(f==-1)
        {
            f=0;
        }
        r=((r+1)%MAX);
        queue[r]=data;
    }
}
void dequeue()
{
    if(f==-1 )
    {
        printf("empty");
        return;
    }
     if (f == r) {

        f = r = -1;}
```

Compilation successful.

```
22  void dequeue()
29      if (f == r) {
32     else
33     {
34         printf("popped element is %d ",queue[f]);
35          f = (f + 1) % MAX;
36     }
37  }
38  int peek()
39  {
40     if(f==-1)
41     {
42         printf("empty");
43         return -1;
44     }
45     else{
46         return queue[f];
47     }
48
49  }
50  void display()
51  {
52     if(f==-1)
53     {
54         printf("empty");
55         return -1;
56     }
57      printf("emelents are");
58
```

```
     printf("emelents are");

     for (int i = f; ; i =( (i + 1) % MAX)) {

         printf("%d ", queue[i]);
         if (i == r){
             break;}}
             printf("\n");
 }
 int main()
 {
     enqueue(10);
     enqueue(20);
     enqueue(30);
     enqueue(35);
     enqueue(50);
     display();
     printf("Peek: %d\n", peek());
     dequeue();
     dequeue();
     printf("Peek: %d\n", peek());
     enqueue(90);
     display();
     return 0;

 }
```