

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  #define MAX_SIZE 10
5
6  int queue[MAX_SIZE];
7  int front = -1;
8  int rear = -1;
9
10 void enqueue(int x);
11 int dequeue();
12 void display();
13
14 int main(){
15     int op,x;
16     while(1){
17         printf("1.Enqueue\n2.Dequeue\n3.Display\n4.Exit\nEnter Your choice: ");
18         scanf("%d",&op);
19         switch(op){
20             case 1:
21                 if((front == 0 && rear == MAX_SIZE - 1) || (front == rear + 1)){
22                     printf("Overflow.\n");
23                 }else{
24                     printf("Enter the item: ");
25                     scanf("%d",&x);
26                     enqueue(x);
27                 }
28                 break;
29             case 2:
30                 if(front == -1){
31                     printf("Underflow.\n");
32                 }else{
33                     x = dequeue();
34                     printf("Removed %d from queue.\n",x);
35                 }
36                 break;
37             case 3:
38                 if(front == -1){
39                     printf("Queue is empty.\n");
40                 }else{
41                     printf("The elements of queue are: ");
42                     display();
43                 }
44                 break;
45             case 4:
46                 exit(0);
47             default:
48                 printf("Invalid Input.\n");
49         }
50     }
51     return 0;
52 }
53
54 void enqueue(int x){
55     if(front == -1){
56         front = 0;
57     }
58     rear = (rear + 1) % MAX_SIZE;
59     queue[rear] = x;
60 }
61
62 int dequeue(){
63     int x = queue[front];
64     if(front == rear){
65         front = -1;
66         rear = -1;
67     }else{
68         front = (front + 1) % MAX_SIZE;
69     }
70     return x;
71 }
72

```

```

73 void display(){
74     int i;
75     if(front <= rear){
76         for(i = front; i <= rear; i++){
77             printf("%d -> ",queue[i]);
78         }
79     }else{
80         for(i = front; i < MAX_SIZE; i++){
81             printf("%d -> ",queue[i]);
82         }
83         for(i = 0; i <= rear; i++){
84             printf("%d -> ",queue[i]);
85         }
86     }
87     printf("NULL\n");
88 }
89
90 /*OUTPUT
91 PS S:\Workspace\CollegeWork\DataStructure> ./a
92 1.Enqueue
93 2.Dequeue
94 3.Display
95 4.Exit
96 Enter Your choice: 1
97 Enter the item: 12
98 1.Enqueue
99 2.Dequeue
100 3.Display
101 4.Exit
102 Enter Your choice: 1
103 Enter the item: 13
104 1.Enqueue
105 2.Dequeue
106 3.Display
107 4.Exit
108 Enter Your choice: 1
109 Enter the item: 14
110 1.Enqueue
111 2.Dequeue
112 3.Display
113 4.Exit
114 Enter Your choice: 3
115 The elements of queue are: 12 -> 13 -> 14 -> NULL
116 1.Enqueue
117 2.Dequeue
118 3.Display
119 4.Exit
120 Enter Your choice: 2
121 Removed 12 from queue.
122 1.Enqueue
123 2.Dequeue
124 3.Display
125 4.Exit
126 Enter Your choice: 2
127 Removed 13 from queue.
128 1.Enqueue
129 2.Dequeue
130 3.Display
131 4.Exit
132 Enter Your choice: 3
133 The elements of queue are: 14 -> NULL
134 1.Enqueue
135 2.Dequeue
136 3.Display
137 4.Exit
138 Enter Your choice: 4
139 PS S:\Workspace\CollegeWork\DataStructure>*/

```