

```

1  #include<stdio.h>
2  #define q_size 5
3  int item,front=-1,rear=-1,q[10];
4  void insert_rear()
5  {
6      if(rear==q_size-1)
7      {
8          printf("queue overflow\n");
9          return;
10     }
11     rear=rear+1;
12     q[rear]=item;
13 }
14 int delete_front()
15 {
16     if(front>rear)
17     {
18         front=0;
19         rear=-1;
20     }
21     return q[front++];
22 }

```

```

23 void displayQ()
24 {
25     int i;
26     if(front>rear)
27         printf("queue empty\n");
28     else
29         for(i=front;i<=rear;i++)
30         {
31             printf("%d\n",q[i]);
32         }
33 }
34 void main()
35 {
36     int choice;
37
38     for(;;)
39     {
40         printf("\n1.insertrear\n2.deletefront\n3.display\n4.exit\n");
41         printf("enter the choice\n");
42         scanf("%d",&choice);
43         switch(choice)
44         {
45             case 1: printf("enter the item to be inserted\n");
46                     scanf("%d",&item);
47                     insert_rear();
48                     break;
49             case 2:
50                 item=delete_front();
51                 if(item==1)

```

```

50                 item=delete_front();
51                 if(item==1)
52                     printf("empty queue\n");
53                 else
54                     printf("element deleted: %d",item);
55                 break;
56             case 3:
57                 displayQ();
58                 break;
59             default:
60                 printf("end of operation\n");
61
62         }
63     }
64 }
65
66 }
67

```

```
1.insertrear
2.deletefront
3.display
4.exit
enter the choice
1
enter the item to be inserted
10

1.insertrear
2.deletefront
3.display
4.exit
enter the choice
1
enter the item to be inserted
20

1.insertrear
2.deletefront
3.display
4.exit
enter the choice
1
enter the item to be inserted
30

1.insertrear
2.deletefront
3.display
```

```
enter the choice
3
0
10
20
30
40
50

1.insertrear
2.deletefront
3.display
4.exit
enter the choice
2
element deleted: 0
1.insertrear
2.deletefront
3.display
4.exit
enter the choice
2
element deleted: 10
1.insertrear
2.deletefront
3.display
4.exit
enter the choice
2
element deleted: 20
1.insertrear
```

```

1.insertrear
2.deletefront
3.display
4.exit
enter the choice
2
element deleted: 20
1.insertrear
2.deletefront
3.display
4.exit
enter the choice
3
30
40
50

1.insertrear
2.deletefront
3.display
4.exit
enter the choice
4
end of operation

1.insertrear
2.deletefront
3.display
4.exit

```

```

1  #include<stdio.h>
2  #include<stdlib.h>
3  # define q_size 5
4  int item,front=0,rear=-1,q[q_size],count=0;
5  void insert_rear()
6  {
7      if(count==q_size)
8      {
9          printf("queue underflow\n");
10         return;
11     }
12     rear=(rear+1)%q_size;
13     q[rear]=item;
14     count++;
15 }
16 int delete_front()
17 {
18     if(count==0)
19         return -1;
20     item=q[front];
21     front=(front+1)%q_size;
22     count=count-1;
23     return item;
24 }
25 void displayQ()
26 {
27     int i,f;
28     if(count==0)
29     {
30         printf("queue is empty\n");
31         return;
32     }
33 }

```

```

32     }
33     f=front;
34     printf("contents of queue\n");
35     for(i=1;i<=count;i++)
36     {
37         printf("%d\n",q[f]);
38         f=(f+1)%q_size;
39     }
40 }
41 void main()
42 {
43     int choice;
44
45     for(;;)
46     {
47         printf("\n1.insertrear\n2.deletefront\n3.display\n4.exit\n");
48         scanf("%d",&choice);
49         switch(choice)
50         {
51             case 1: printf("enter the item to be inserted\n");
52                     scanf("%d",&item);
53                     insert_rear();
54                     break;
55             case 2: item=delete_front();
56                     if(item==-1)
57                         printf("\n empty queue");
58                     else
59                         printf("item deleted=%d\n",item);
60                     break;
61             case 3:
62                     displayQ();

```

```

61             case 3:
62                 displayQ();
63                 break;
64             default:
65                 printf("enf of operation");
66                 exit(0);
67         }
68     }
69 }
70 }
71 }
72 }
73 }
74 }

```

```

1.insertrear
2.deletefront
3.display
4.exit
1
enter the item to be inserted
1

1.insertrear
2.deletefront
3.display
4.exit
1
enter the item to be inserted
3

1.insertrear
2.deletefront
3.display
4.exit
1
enter the item to be inserted
5

```

```
enter the item to be inserted
5

1.insertrear
2.deletefront
3.display
4.exit
1
enter the item to be inserted
7

1.insertrear
2.deletefront
3.display
4.exit
1
enter the item to be inserted
9

1.insertrear
2.deletefront
```

```
1.insertrear
2.deletefront
3.display
4.exit
3
contents of queue
1
3
5
7
9

1.insertrear
2.deletefront
3.display
4.exit
2
item deleted=1

1.insertrear
2.deletefront
3.display
```

```
2.deletefront
3.display
4.exit
2
item deleted=3

1.insertrear
2.deletefront
3.display
4.exit
3
contents of queue
5
7
9

1.insertrear
2.deletefront
3.display
4.exit
4
enf of operation
```