

Queue

```
#include <stdio.h>
#include <conio.h>
#include <process.h>
#define QUE_SIZE 3
```

```
int item, front = 0, rear = -1, q[10];
```

```
void insertrear()
```

```
{
    if (rear == QUE_SIZE - 1)
    {
        printf("Queue overflow\n");
        return;
    }
}
```

```
{
    rear = rear + 1;
    q[rear] = item;
}
```

```
void deletefront() {
    if (front > rear)
    {
```

```
        front = 0;
        rear = -1;
        return 1;
    }
```

```
{
    return q[front++];
}
```

```
}
```

```
void display()
```

```
{
```

```
    int i;
```

```
    if (front > rear)
```

```
        printf("Queue is empty");
```

```
    return;
```

```
}
```

```
else
```

```
    printf("Contents of queue\n");
```

```
    for (i = front; i <= rear; i++)
```

```
        printf("%d\n", q[i]);
```

```
}
```

```
}
```

```
void main()
```

```
{
```

```
    int choice;
```

```
    for (i = 1; i <= 4; i++)
```

```
        printf("%d 1: insert rear\n 2: delete front\n
```

```
  3: display\n  4: exit\n");
```

```
    printf("Enter the choice\n");
```

```
    scanf("%d", &choice);
```

```
    switch (choice)
```

```
    {  
        case 1: printf("Enter the item to be
```

```
        inserted\n");
```

```
        scanf("%d", &item);
```

```
        insert_rear();
```

break;

Case 2: item = delete front();

if (item == -1)

printf("queue is empty\n");

else

printf("item deleted = %d\n", item);

break;

Case 3: display Q L)

break;

default : exit(0);

```

1  #include <stdio.h>
2  #include<conio.h>
3  #include<process.h>
4  #define QUE_SIZE 3
5  int item,front=0,rear=-1,q[10];
6  void insertrear()
7  {
8      if(rear==QUE_SIZE-1)
9      {
10         printf("Queue overflow\n");
11         return;
12     }
13     rear=rear+1;
14     q[rear]=item;
15 }
16 int deletefront(){
17     if(front>rear)
18     {
19         front=0;
20         rear=-1;
21         return -1;
22     }
23     return q[front++];
24 }

```

```

25 void displayQ()
26 {
27     int i;
28     if(front>rear){
29         printf("Queue is empty");
30         return;
31     }
32     else
33     printf("contents of queue\n");
34     for(i=front;i<=rear;i++){
35         printf("%d\n",q[i]);
36     }
37 }
38 void main()
39 {
40     int choice;
41     for(;;){
42         printf("\n 1:insert rear\n 2: delete front\n 3:display\n 4:exit\n");
43         printf("Enter the choice\n");
44         scanf("%d",&choice);
45         switch(choice){
46             case 1:printf("Enter the item to be inserted\n");
47                 scanf("%d",&item);
48                 insertrear();

```

```

40 int choice;
41 for(;;){
42     printf("\n 1:insert rear\n 2: delete front\n 3:display\n 4:exit\n");
43     printf("Enter the choice\n");
44     scanf("%d",&choice);
45     switch(choice){
46         case 1:printf("Enter the item to be inserted\n");
47             scanf("%d",&item);
48             insertrear();
49             break;
50         case 2:item=deletefront();
51         if(item== -1)
52             printf("queue is empty\n");
53         else
54             printf("item deleted=%d\n",item);
55         break;
56         case 3:displayQ();
57         break;
58         default:exit(0);
59     }
60 }
61
62
63

```

```
1:insert rear
2: delete front
3:display
4:exit
Enter the choice
1
Enter the item to be inserted
10

1:insert rear
2: delete front
3:display
4:exit
Enter the choice
1
Enter the item to be inserted
20

1:insert rear
2: delete front
3:display
4:exit
Enter the choice
1
Enter the item to be inserted
30

1:insert rear
2: delete front
```

```
1:insert rear
2: delete front
3:display
4:exit
Enter the choice
1
Enter the item to be inserted
40
Queue overflow
```

```
1:insert rear
2: delete front
3:display
4:exit
Enter the choice
3
contents of queue
10
20
30
```

```
1:insert rear
2: delete front
3:display
4:exit
Enter the choice
2
item deleted=10
```


30

```
1:insert rear
2: delete front
3:display
4:exit
```

Enter the choice

```
2
item deleted=10
```

```
1:insert rear
2: delete front
3:display
4:exit
```

Enter the choice

```
2
item deleted=20
```

```
1:insert rear
2: delete front
3:display
4:exit
```

Enter the choice

```
2
item deleted=30
```

```
1:insert rear
2: delete front
3:display
4:exit
```

```
3:display
4:exit
Enter the choice
2
item deleted=30

1:insert rear
2: delete front
3:display
4:exit
Enter the choice
2
queue is empty

1:insert rear
2: delete front
3:display
4:exit
Enter the choice
```