

```

Start here X linear.c X circular.c X
1  #include<stdio.h>
2  #include<stdlib.h>
3  #define QUE_SIZE 3
4  int item,front=0,rear=-1,q[10];
5  void insertrear()
6  {if(rear==QUE_SIZE-1)
7  {
8      printf("queue overflow\n");
9      return;
10 }
11 rear=rear+1;
12 q[rear]=item;
13 }
14 int deletefront()
15 {if (front>rear)
16 {front=0;
17 rear=-1;
18 return -1;
19 }return q[front++];
20 }
21 void displayQ()
22 {int i;
23 if (front>rear)
24 {
25     printf("queue is empty\n");
26     return;
27 }
28 printf("contents of queue\n");
29 for(i=front;i<=rear;i++)
30 {
31     printf("%d\n",q[i]);
32 }
33 }
34 int main()
35 {
36     int choice;
37     for(;;)
38     {
39         printf("1:insertrear 2:deletefront 3:display 4:exit\n");
40         printf("enter the choice\n");
41         scanf("%d",&choice);
42         switch(choice)
43         {
44             case 1:printf("enter the item to be inserted\n");
45                     scanf("%d",&item);

```

```

30 printf("queue is empty\n");
31 return;
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)%QUE_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 printf("enter the choice\n");
49 scanf("%d",&choice);
50
51 switch(choice)
52 {
53 {
54 case 1:printf("enter the item to be inserted\n");
55 scanf("%d",&item);
56 insertrear();
57 break;
58 case 2:item=deletefront();
59 if(item==1)
60 printf("queue is empty\n");
61 else
62 printf("item deleted =%d\n",item);
63 break;
64 case 3:displayQ();
65 break;
66 default:exit(0);
67 }
68 }
69 getch();
70 }
71 }

```

logs & others

Code-Retrieval X Search results X C++ Build log X Build messages X C++ Check/Verify X C++ Check/Verify messages X Corona X Delusion X Download X Fortran info X C++ Check file list X Thread search X
 C/C++ Windows (CR+LF) WINDOWS 1252 Line 43 Col 13 Row 602 build Build file list

```
3
1:insertrear
2:deletefront
3:display
4:exit
enter the choice
1
enter the item to be inserted
4
queue overflow
1:insertrear
2:deletefront
3:display
4:exit
enter the choice
2
item deleted =2
1:insertrear
2:deletefront
3:display
4:exit
enter the choice
7
item deleted =2
1:insertrear
2:deletefront
3:display
4:exit
enter the choice
2
item deleted =3
1:insertrear
2:deletefront
3:display
4:exit
enter the choice
2
queue is empty
1:insertrear
2:deletefront
3:display
4:exit
enter the choice
```

```
1:insertrear
2:deletefront
3:display
4:exit
enter the choice
2
item deleted =2

1:insertrear
2:deletefront
3:display
4:exit
enter the choice
2
item deleted =2

1:insertrear
2:deletefront
3:display
4:exit
enter the choice
2
item deleted =3

1:insertrear
2:deletefront
3:display
4:exit
enter the choice
2
queue is empty

1:insertrear
2:deletefront
3:display
4:exit
enter the choice
3
queue is empty

1:insertrear
2:deletefront
3:display
4:exit
enter the choice
4

Process returned 0 (0x0)   execution time : 66.843 s
Press any key to continue.
```

```

:insertrear
:deletefront
:display
::exit
enter the choice

enter the item to be inserted

:insertrear
:deletefront
:display
::exit
enter the choice

enter the item to be inserted

:insertrear
:deletefront
:display
::exit
enter the choice

enter the item to be inserted

queue overflow

:insertrear
:deletefront
:display
::exit
enter the choice

```