

Display:-

- 1) start
- 2) check if front == -1
 print queue is empty and stop
- 3) move front to rear
 print queue[i]
- 4) stop

```
#include <stdio.h>
#include <ctype.h>
#include <string.h>
#define n 5
```

```
int queue (int x)
{
    int front = -1;
    int rear = -1;
```

```
void enqueue (int x)
{
```

```
    if (rear == n-1)
```

```
    {
        printf("queue overflow");
    }
```

```
    else if (front == -1 && rear == -1)
```

```
    {
        front = rear = 0;
```

```
        queue[rear] = x;
```

```
    }
    else {
```

```
        rear++;
```

```
        queue[rear] = x;
```

```
    }
}
```

output:

Queue operations

DATE:

PAGE:

1. enqueue

2. dequeue

3. display

Enter your choice: 1

Enter your element to insert: 34

Enter your choice: 1

Enter your element to insert: 23

Enter your choice: 3

34 23

Enter your choice: 1

Enter your element to insert: 45

Enter your choice: 1

Enter your element to insert: 46

Queue overflow

Enter your choice: 3

~~34 23 45 46~~

Enter your choice: 2

Element is deleted

13/10/21

```
void main ()  
{
```

```
    int ch;
```

```
    while (1) {
```

```
        printf("\n queue operations:\n");
```

```
        printf("1. enqueue\n");
```

```
        printf("2. dequeue\n");
```

```
        printf("3. display\n");
```

```
        printf("Enter your choice:");
```

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

```
        switch (ch)
```

```
        {
```

```
            case 1:
```

```
                printf("enter a element to insert:");
```

```
                int x;
```

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

```
                enqueue(x);
```

```
                break;
```

```
            case 2: dequeue();
```

```
                break;
```

```
            case 3: display();
```

```
                break;
```

```
            default:
```

```
                printf("\n invalid choice\n");
```

```
                break;
```

```
        }
```

```
    }
```

```
    return 0;
```

```
}
```

void dequeue()

{
if (front == -1 && rear == -1)

{ printf("queue is empty");

else if (front == rear)

{ front = rear = -1;

}

else {

printf("deleted element: ", queue[front]);

front++;

}

void display()

{
int i;

if (front == -1 && rear == -1)

{ ~~printf~~ printf("queue is empty");

}

else {

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

{ printf("%d\t", queue[i]);

}

}

}

```

1 #include<stdio.h>
2 #include<ctype.h>
3 #include<string.h>
4 #define n 5
5 int queue[n];
6 int front=-1;
7 int rear=-1;
8
9 void enqueue(int x)
10 {
11     if(rear==n-1)
12     {
13         printf("Queue overflow");
14     }
15     else if(front==0 && rear==n-1)
16     {
17         front=rear+1;
18         queue[rear]=x;
19     }
20     else
21     {
22         rear++;
23         queue[rear]=x;
24     }
25 }
26 void dequeue()
27 {
28     if(front==0 && rear==n-1)
29     {
30         printf("Queue is empty");
31     }
32     else if (front==rear)
33     {
34         front=rear-1;
35     }
36     else
37     {
38         printf("Deleted element",queue[front]);
39         front++;
40     }
41 }
42 void display()
43 {
44     if (front==0 && rear==n-1)
45     {
46         printf("Queue is empty");
47     }
48     else
49     {
50         for (i=front; i<=rear; i++)
51         {
52             printf("%d\t",queue[i]);
53         }
54     }
55 }
56 int main()
57 {
58     int ch;
59 }

```

1. enqueue
2. Dequeue
3. Display
Enter your choice: 1
Enter a elements to insert:234
Queue Operations:
1. enqueue
2. Dequeue
3. Display
Enter your choice: 3
234
Queue Operations:
1. enqueue
2. Dequeue
3. Display
Enter your choice: 2
Deleted element=
Queue Operations:
1. enqueue
2. Dequeue
3. Display
Enter your choice: 3
234
Queue Operations:
1. enqueue
2. Dequeue
3. Display
Enter your choice: 1
Enter a elements to insert:34
Queue Operations:
1. enqueue
2. Dequeue
3. Display
Enter your choice: 1
Enter a elements to insert:56
Queue Operations:
1. enqueue
2. Dequeue
3. Display
Enter your choice: 1
Enter a elements to insert:56
Queue overflow
Queue Operations:
1. enqueue
2. Dequeue
3. Display
Enter your choice: 2
Deleted element=
Queue Operations:
1. enqueue
2. Dequeue
3. Display
Enter your choice:

```
gnumc - CodeBlocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins Dcc/Blocks Settings Help
Start here X: "queue.c" X
37 }
38 }
39 void display()
40 {
41     if (front==0 && rear==0)
42     {
43         printf("queue is empty");
44     }
45     else
46     {
47         for (i=front; i<=rear; i++)
48         {
49             printf("%d\t", queue[i]);
50         }
51     }
52 }
53 void main()
54 {
55     int ch;
56     while (1)
57     {
58         printf("\nQueue Operations:\n");
59         printf("1. enqueue\n");
60         printf("2. dequeue\n");
61         printf("3. Display\n");
62         printf("Enter your choice: ");
63         scanf("%d", &ch);
64         switch (ch)
65         {
66             case 1:
67                 printf("Enter a element to insert:");
68                 int x;
69                 scanf("%d", &x);
70                 enqueue(x);
71                 break;
72             case 2:
73                 dequeue();
74                 break;
75             case 3:
76                 display();
77                 break;
78             default:
79                 printf("Invalid choice\n");
80                 break;
81         }
82     }
83     return 0;
84 }
85 }
86 }
87 }
88 }
89 }
```

D:\chethan\CPP\queue.c C/C++ Windows (C++11) WINDOWS-1252 Line 74, Col 18, Row 135 Insert Modified Exec/Write default 98.34 13-10-2023