

Display:-

- 1) start
- 2) check if front == -1  
print queue is empty and stop
- 3) move front to rear.  
print queue[1]
- 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 += 1;
```

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

```
}
```

Output:

Queue Operations

1. enqueue
2. dequeue
3. display

Enter your choice: 1

Enter your element to insert: 34

Enter your choice: 2

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: 2

Enter your element to insert: 65

Queue overflow

Enter your choice: 3

34 23 45 65

Enter your choice: 2

Element is deleted

3/10/18

```
void main ()
```

```
{  
    int ch;  
    while (1) {  
        printf ("In 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 elements to insert: ");  
    int x;  
    scanf ("%d", &x);  
    enqueue (x);  
    break;
```

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

```
break;
```

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

```
break;
```

```
default:
```

```
    printf ("In 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("queue is empty");
}
else {
    for (i=front; i<=rear; i++)
    {
        printf("\t%d", queue[i]);
    }
}
```

queue.c - Code::Blocks 20.03

File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins Doxygen Settings Help

Start here X "queue.c" X

```

1 //include<stdio.h>
2 //include<conio.h>
3 //include<string.h>
4 #define n 5
5 int front=-1;
6 int rear=-1;
7 int arr[n];
8
9 void enqueue(int x)
10 {
11     if(rear==n-1)
12     {
13         printf("Queue overflow");
14     }
15     else if(front==n-1 && rear==n-1)
16     {
17         front=rear=0;
18         queue[rear]=x;
19     }
20     else
21     {
22         rear++;
23         queue[rear]=x;
24     }
25 }
26 void dequeue()
27 {
28     if(front==n-1 && 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 elements",queue[front]);
39         front++;
40     }
41 }
42 void display()
43 {
44     int i;
45     if(front==n-1 && rear==n-1)
46     {
47         printf("Queue is empty");
48     }
49     else
50     {
51         for (i=front; i<=rear;i++)
52         {
53             printf("%d",queue[i]);
54         }
55     }
56 }
57 void main()
58 {
59     int ch;
60 }
```

D:\chethan\DP\Queue.c C/C++ Windows (CR+LF) WINDOWS-1252 Line 35, Col 33, Pos 569 Insert Modified Read/Write default ENG IN 09:34 13-10-2023

D:\chethan\DP\Queue.c

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  
Selected elements:  
Queue Operations:  
1. enqueue  
2. Dequeue  
3. Display  
Enter your choice: 3  
3  
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: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  
Selected elements:  
Queue Operations:  
1. enqueue  
2. Dequeue  
3. Display  
Enter your choice: -

```
#include <stdio.h>
#include <conio.h>

void display()
{
    int i;
    if (front==0 && rear==0)
        printf("Queue is empty");
    else
        for (i=front; i<=rear;i++)
            printf("%d", queue[i]);
}

void main()
{
    int ch;
    while (1)
    {
        printf("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 elements to insert:\n");
                int x;
                scanf("%d", &x);
                enqueue(x);
                break;
            case 2:
                dequeue();
                break;
            case 3:
                display();
                break;
            default:
                printf("An invalid choice\n");
                break;
        }
    }
    return 0;
}
```

D:\chethan\DP\queue.c

C/C++ Windows (CR+LF) WINDOWS-1252 Line 74, Col 19, Pos 1351

Insert Modified Read/Write default

^ ENG IN 09:34 13-10-2023