

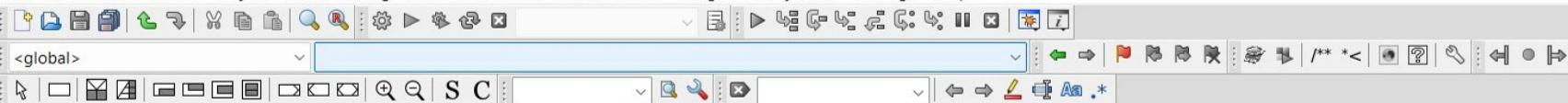
Start here x Single linked list Stack operation.c x Single linked List Queue operation.c x

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  struct Node {
5      int data;
6      struct Node* next;
7  };
8
9  void enqueue(struct Node** front, struct Node** rear, int value) {
10     struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
11     newNode->data = value;
12     newNode->next = NULL;
13
14     if (*rear == NULL) {
15         *front = *rear = newNode;
16     } else {
17         (*rear)->next = newNode;
18         *rear = newNode;
19     }
20     printf("%d enqueued to queue\n", value);
21 }
22
23 int dequeue(struct Node** front, struct Node** rear) {
24     if (*front == NULL) {
25         printf("Queue Underflow\n");
26         return -1;
27     }
28     struct Node* temp = *front;
29     int value = temp->data;
30
31     *front = (*front)->next;
32     if (*front == NULL)
33         *rear = NULL;
34
35     free(temp);
36     return value;
37 }
38
39 void display(struct Node* front) {
```



Start here x Single linked list Stack operation.c x Single linked List Queue operation.c x

```
40 if (front == NULL) {
41     printf("Queue is empty\n");
42     return;
43 }
44 struct Node* temp = front;
45 printf("Queue: ");
46 while (temp != NULL) {
47     printf("%d -> ", temp->data);
48     temp = temp->next;
49 }
50 printf("NULL\n");
51 }
52
53 int main() {
54     struct Node *front = NULL, *rear = NULL;
55     int choice, value;
56
57     while (1) {
58         printf("\nQUEUE MENU\n");
59         printf("1. Enqueue\n");
60         printf("2. Dequeue\n");
61         printf("3. Display\n");
62         printf("4. Exit\n");
63         printf("Enter your choice: ");
64         scanf("%d", &choice);
65
66         switch (choice) {
67             case 1:
68                 printf("Enter value: ");
69                 scanf("%d", &value);
70                 enqueue(&front, &rear, value);
71                 break;
72             case 2:
73                 value = dequeue(&front, &rear);
74                 if (value != -1)
75                     printf("Dequeued element: %d\n", value);
76                 break;
77             case 3:
78                 display(front);
```



```
49     }
50     printf("NULL\n");
51 }
52
53 int main() {
54     struct Node *front = NULL, *rear = NULL;
55     int choice, value;
56
57     while (1) {
58         printf("\nQUEUE MENU\n");
59         printf("1. Enqueue\n");
60         printf("2. Dequeue\n");
61         printf("3. Display\n");
62         printf("4. Exit\n");
63         printf("Enter your choice: ");
64         scanf("%d", &choice);
65
66         switch (choice) {
67             case 1:
68                 printf("Enter value: ");
69                 scanf("%d", &value);
70                 enqueue(&front, &rear, value);
71                 break;
72             case 2:
73                 value = dequeue(&front, &rear);
74                 if (value != -1)
75                     printf("Dequeued element: %d\n", value);
76                 break;
77             case 3:
78                 display(front);
79                 break;
80             case 4:
81                 exit(0);
82             default:
83                 printf("Invalid choice\n");
84         }
85     }
86 }
87
```

```
QUEUE MENU
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 2
Queue Underflow
```

```
QUEUE MENU
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 1
Enter value: 45
45 enqueued to queue
```

```
QUEUE MENU
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 1
Enter value: 18
18 enqueued to queue
```

```
QUEUE MENU
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 1
Enter value: 99
99 enqueued to queue
```

```
QUEUE MENU
1. Enqueue
2. Dequeue
3. Display
4. Exit
```

```
"E:\Monisha\BMSCE\SEM-III\I  +  -  X
QUEUE MENU
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 1
Enter value: 73
73 enqueued to queue

QUEUE MENU
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 3
Queue: 45 -> 18 -> 99 -> 73 -> NULL

QUEUE MENU
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 2
Dequeued element: 45

QUEUE MENU
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 3
Queue: 18 -> 99 -> 73 -> NULL

QUEUE MENU
1. Enqueue
2. Dequeue
3. Display
4. Exit
Enter your choice: 4

Process returned 0 (0x0)    execution time : 69.442 s
```