

C stacklist.c > display()

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  struct node
5  {
6      int info;
7      struct node *ptr;
8  } *top, *top1, *temp;
9
10 int count = 0;
11 void push(int data)
12 {
13     if (top == NULL)
14     {
15         top = (struct node *)malloc(1 * sizeof(struct node));
16         top->ptr = NULL;
17         top->info = data;
18     }
19     else
20     {
21         temp = (struct node *)malloc(1 * sizeof(struct node));
22         temp->ptr = top;
23         temp->info = data;
24         top = temp;
25     }
26     count++;
27     printf("Node is Inserted\n");
28 }
29 int pop()
30 {
31     top1 = top;
32     if (top1 == NULL)
33     {
34         printf("Stack Underflow\n");
35         return -1;
36     }
37     else
38     {
39         top1 = top1->ptr;
40         int popped = top->info;
41         free(top);
42         top = top1;
43         count--;
44         return popped;
45     }
46 }
```

C stacklist.c > display()

```
45 void display()
46 {
47     top1 = top;
48     if (top1 == NULL)
49     {
50         printf("Stack Underflow\n");
51         return;
52     }
53     printf("The stack is: ");
54     while (top1 != NULL)
55     {
56         printf("%d-->", top1->info);
57         top1 = top1->ptr;
58     }
59     printf("NULL\n");
60 }
61
62 int main()
63 {
64     int choice, value;
65     while (1)
66     {
67         printf("\n1.Push\n2.Pop\n3.Display\n4.Exit\n");
68         printf("Enter your choice: ");
69         scanf("%d", &choice);
70         switch (choice)
71         {
72             case 1:
73                 printf("Enter the value to insert: ");
74                 scanf("%d", &value);
75                 push(value);
76                 break;
77             case 2:
78                 printf("Popped element is: %d", pop());
79                 break;
80             case 3:
81                 display();
82                 break;
83             case 4:
84                 exit(0);
85                 break;
86             default:
87                 printf("Wrong choice");
88         }
89     }
90 }
```

1.Push
2.Pop
3.Display
4.Exit
Enter your choice: 1
Enter the value to insert: 12
Node is Inserted

1.Push
2.Pop
3.Display
4.Exit
Enter your choice: 1
Enter the value to insert: 23
Node is Inserted

1.Push
2.Pop
3.Display
4.Exit
Enter your choice: 1
Enter the value to insert: 39
Node is Inserted

1.Push
2.Pop
3.Display
4.Exit
Enter your choice: 1
Enter the value to insert: 41
Node is Inserted

1.Push
2.Pop
3.Display
4.Exit
Enter your choice: 3
The stack is: 41-->39-->23-->12-->NULL

1.Push
2.Pop
3.Display
4.Exit
Enter your choice: 2
Popped element is: 41
1.Push
2.Pop
3.Display
4.Exit
Enter your choice: 3
The stack is: 39-->23-->12-->NULL

1.Push
2.Pop
3.Display
4.Exit
Enter your choice: 4

queue.c > insert(node *, int)

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

struct node
{
    int data;
    struct node *next;
};

struct node *front = NULL;
struct node *rear = NULL;

void insert(struct node *ptr, int item){
    ptr = (struct node *) malloc(sizeof(struct node));
    if(ptr == NULL){
        printf("Overflow\n");
        return ;
    }
    else {
        ptr->data = item;
        if(front == NULL){
            front = ptr;
            rear = ptr;
            front->next = NULL;
            rear->next = NULL;
        }else {
            rear->next = ptr;
            rear = ptr;
            rear->next = NULL;
        }
    }
}

void deleteNode(struct node*ptr){
    if(front == NULL){
        rear = NULL;
        printf("Underflow\n");
        return ;
    }
    else {
        ptr = front;
        front = front->next;
        printf("%d is deleted\n",ptr->data);
        free(ptr);
    }
}
```



```
44 }
45 void display(){
46     struct node *ptr;
47     ptr = front;
48     if(front == NULL && rear == NULL){
49         printf("Queue is empty\n");
50     }else {
51         while(ptr->next != NULL){
52             printf("%d-->",ptr->data);
53             ptr = ptr->next;
54         }
55         printf("%d\n",ptr->data);
56     }
57 }
58 int main(){
59     struct node * head = NULL;
60     int choice, value;
61     printf("\n1.Insert\n2.Delete\n3.Display\n4.Exit\n");
62     while(1){
63         printf("Enter your choice: ");
64         scanf("%d",&choice);
65         switch(choice){
66             case 1: printf("Enter the value to be inserted: ");
67                     scanf("%d",&value);
68                     insert(head,value);
69                     break;
70             case 2: deleteNode(head);
71                     break;
72             case 3: display();
73                     break;
74             case 4: exit(0);
75         }
76     }
77 }
78
79
```

PS C:\Users\gsm22\OneDrive\Documents\DS> cd "c:\Users\gsm22\OneDrive\Documents\DS" & gcc queue.c -o queue ; if (\$?) { .\queue }

1.Insert

2.Delete

3.Display

4.Exit

Enter your choice: 1

Enter the value to be inserted: 12

Enter your choice: 1

Enter the value to be inserted: 45

Enter your choice: 1

Enter the value to be inserted: 35

Enter your choice: 1

Enter the value to be inserted: 86

Enter your choice: 3

12-->45-->35-->86

Enter your choice: 2

12 is deleted

Enter your choice: 3

45-->35-->86

Enter your choice: 2

45 is deleted

Enter your choice: 3

35-->86

Enter your choice: 4

PS C:\Users\gsm22\OneDrive\Documents\DS> █