

PRN: 2020BTEIT00041

Stack Implementation using LinkedList:

```

1  /*
2  |   Name: Om Vivek Gharge
3  |   PRN: 2020BTEIT00041
4  */
5
6  // Stack Implementation Using LinkedList :
7
8  #include <bits/stdc++.h>
9  using namespace std;
10 typedef long long ll;
11
12 class Node{
13 public:
14     int data,
15         size;
16     Node* top = NULL;
17     Node* next;
18
19     Node(int data){
20         this->data = data;
21         this->next = NULL;
22     }
23 };
24
25 class Head{
26 public:
27     int count;
28     Node* head = NULL;
29
30     Head(){
31         this->count = 0;
32         this->head = NULL;
33     }
34 };
35
36 //Push
37 void Push(Head* h, int data){
38     // creating a new node
39     Node* new_node = new Node(data);
40
41     // checking overflow condition
42     if(h->count > new_node->size) cout<<"Stack Overflow\n";
43
44     else{
45         // if stack is empty
46         if(h->count == 0){
47             h->head = new_node;
48
49             h->count++;

```

```

50         new_node->top = h->head;
51     }
52     else{
53         new_node->next = h->head;
54         h->head = new_node;
55
56         h->count++;
57         new_node->top = h->head;
58     }
59 }
60 }
61
62 //Pop
63 void Pop(Head* h){
64
65     // Underflow condition check
66     if(h->count == 0) cout<<"Stack Underflow\n";
67     else{
68         Node* del_node = h->head;
69
70         h->head = h->head->next;
71         del_node->next = NULL;
72         h->count--;
73
74         cout<<"Popped element is "<<del_node->data<<endl;
75         free(del_node);
76     }
77 }
78
79 //Peek
80 void Peek(Head* h){
81     if(h->count == 0) cout<<"Stack Underflow\n";
82     else{
83         cout<<"Top element is "<<h->head->data<<endl;
84     }
85 }
86
87 //Display
88 void Display(Head* h){
89     // Underflow condition check
90     if(h->count == 0) cout<<"Stack Underflow\n";
91     else{
92         // Traversing the stack
93         Node* p = h->head;
94         for(int i=0; i<h->count; i++){
95             cout<<p->data<<" ";

```

```

96         p = p->next;
97     }
98     cout<<endl;
99 }
100 }
101
102 int main(){
103     Head* h = new Head();
104
105     // Menu driven program to perform stack operations
106     int choice;
107     do{
108         cout<<"\n1. Push\n2. Pop\n3. Peek\n4. Display\n5. Exit\n";
109
110         cout<<"Enter your choice: ";
111         cin>>choice;
112
113         switch(choice){
114             case 1:
115                 int data;
116                 cout<<"Enter data to be pushed: ";
117                 cin>>data;
118                 Push(h, data);
119                 break;
120             case 2:
121                 Pop(h);
122                 break;
123             case 3:
124                 Peek(h);
125                 break;
126             case 4:
127                 Display(h);
128                 break;
129             case 5:
130                 cout<<"Exiting...\n";
131                 break;
132             default:
133                 cout<<"Invalid choice\n";
134         }
135     }while(choice != 5);
136
137     return 0;
138 }
139

```

## OUTPUT:

```
1. Push
2. Pop
3. Peek
4. Display
5. Exit
Enter your choice: 1
Enter data to be pushed: 1

1. Push
2. Pop
3. Peek
4. Display
5. Exit
Enter your choice: 1
Enter data to be pushed: 2

1. Push
2. Pop
3. Peek
4. Display
5. Exit
Enter your choice: 1
Enter data to be pushed: 3

1. Push
2. Pop
3. Peek
4. Display
5. Exit
Enter your choice: 1
Enter data to be pushed: 4

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

1. Push
2. Pop
3. Peek
4. Display
5. Exit
Enter your choice: 2
Popped element is 4

1. Push
2. Pop
3. Peek
4. Display
5. Exit
Enter your choice: 5
Exiting...
```