PRN: 2020BTEIT00041

Stack Implementation using LinkedList:

```cpp
/*
    Name: Om Vivek Gharge
    PRN: 2020BTEIT00041
*/

// Stack Implementation Using LinkedList :

#include <bits/stdc++.h>
using namespace std;
typedef long long ll;

class Node{
public:
    int data,
        size;
    Node* top = NULL;
    Node* next;

    Node(int data){
        this->data = data;
        this->next = NULL;
    }
};

class Head{
public:
    int count;
    Node* head = NULL;

    Head(){
        this->count = 0;
        this->head = NULL;
    }
};

//Push
void Push(Head* h, int data){
    // creating a new node
    Node* new_node = new Node(data);

    // checking overflow condition
    if(h->count > new_node->size) cout<<"Stack Overflow\n";

    else{
        // if stack is empty
        if(h->count == 0){
            h->head = new_node;

            h->count++;
```

```
            new_node->top = h->head;
        }
        else{
            new_node->next = h->head;
            h->head = new_node;

            h->count++;
            new_node->top = h->head;
        }
    }
}

//Pop
void Pop(Head* h){

    // Underflow condition check
    if(h->count == 0) cout<<"Stack Underflow\n";
    else{
        Node* del_node = h->head;

        h->head = h->head->next;
        del_node->next = NULL;
        h->count--;

        cout<<"Popped element is "<<del_node->data<<endl;
        free(del_node);
    }
}

//Peek
void Peek(Head* h){
    if(h->count == 0) cout<<"Stack Underflow\n";
    else{
        cout<<"Top element is "<<h->head->data<<endl;
    }
}

//Display
void Display(Head* h){
    // Underflow condition check
    if(h->count == 0) cout<<"Stack Underflow\n";
    else{
        // Traversing the stack
        Node* p = h->head;
        for(int i=0; i<h->count; i++){
            cout<<p->data<<" ";
```

```
            p = p->next;
        }
        cout<<endl;
    }
}

int main(){
    Head* h = new Head();

    // Menu driven program to perform stack operations
    int choice;
    do{
        cout<<"\n1. Push\n2. Pop\n3. Peek\n4. Display\n5. Exit\n";

        cout<<"Enter your choice: ";
        cin>>choice;

        switch(choice){
            case 1:
                int data;
                cout<<"Enter data to be pushed: ";
                cin>>data;
                Push(h, data);
                break;
            case 2:
                Pop(h);
                break;
            case 3:
                Peek(h);
                break;
            case 4:
                Display(h);
                break;
            case 5:
                cout<<"Exiting...\n";
                break;
            default:
                cout<<"Invalid choice\n";
        }
    }while(choice != 5);


    return 0;
}
```

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...
```