```cpp
#include <bits/stdc++.h>
using namespace std;

class Node {

public:
    int value;
    Node*next;
    Node * prev;

    Node (int val){

        value =val;
        next=NULL;
        prev=NULL;
    }
};

class Stack {

    Node * head;
    Node * top;
    int count = 0;

public:

    Stack(){
        head =NULL;
        top = NULL;
    }
void push(int val){
    Node *newNode = new Node (val);

    if (head== NULL){
        head = top = newNode;
        count ++;
        return;
    }

    top->next=newNode;
    newNode->prev = top;
    top =  newNode;
    count++;
}
```

```cpp
int pop (){
    Node * dellNode;
    dellNode=top;
    int chk = -1;
    if (head == NULL){
        cout<<" Stack Undrflow "<<endl;
        return chk;
    }
    if (top == head){
        head = top = NULL;
    }

    else {
        top = dellNode->prev;
        top->next = NULL;
    }

    chk = dellNode->value;
    delete dellNode;
    count --;
    return chk;

}

bool empty (){
    if (head== NULL) return true;
    else return false;
}

int Top(){

    int chk;
    if (top == NULL){
        cout<<" There is no element in the top"<<endl;
        chk=-1;
    }
    else {
        chk = top->value;
        return chk;
    }
}

int size(){
```

```cpp
        return count ;
    }
};

int main() {

    Stack st;


    st.push(1);
    st.push(2);
    st.push(3);


  while (!st.empty()){
   cout<< st.pop()<<endl;
  }
  cout<<st.size()<<endl;

  if (!st.empty())

  {
     //cout<<st.Top<<endl;
  }
    return 0;
}
```