

STACK OF LINKED LIST

Eman T.Mahdi

College of C.S. &I.T.

OVER VIEW

- General definition of stack
- Some operation of stack on L.L.

STACK

A stack is an ordered collection of items (elements), in which the removal and addition of stack items can take place only at the top of the stack.

- To add (push) an item to the stack, it must be placed on the top of the stack.
- To remove (pop) an item from the stack, it must be removed from the top of the stack too.

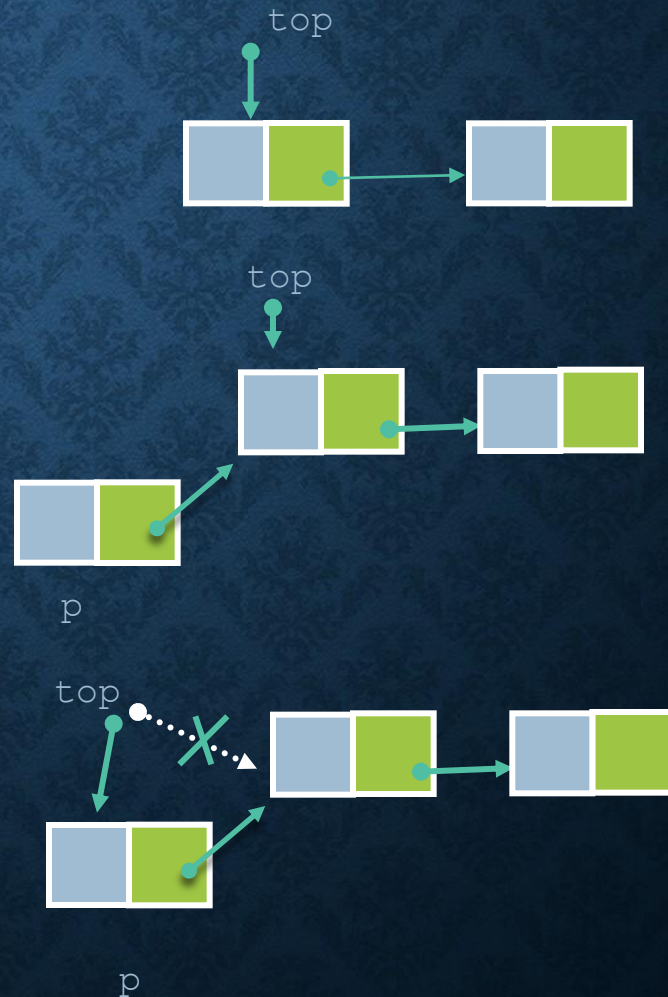
Thus, the last element that is pushed into the stack, is the first element to be popped out of the stack. i.e., Last In First Out (LIFO)

STACK OF LINKED LIST

```
#include<iostream.h>
#include<conio.h>
#include<stdlib.h>
struct node
{
    int data;
    node*next;
};
node*top,*p,*q;
```

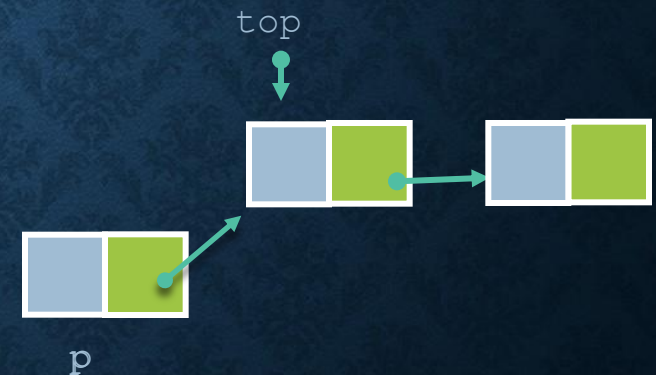
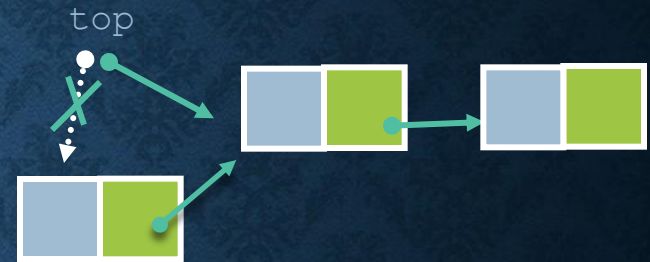

PUSH INTO STACK

```
void push()
{
    p=new node;
    cout<<"input element\n";
    cin>>p->data;
    if(top==NULL)
        p->next=NULL;
    else
        p->next=top;
    top=p;
}
```



POP FROM STACK

```
void pop()
{
    int value;
    if(top==NULL)
    {cout<<"error....linked stack is empty\n";
    cout<<"press any key to exit\n";
    }
    getch();
    exit(·);
    else
    {q=top;
    value=q->data;
    top=q->next;
    q->next=NULL;
    delete(q;
    }}
```



MAIN PROGRAM

```
void main()
{
int choice,l,m,i,item;
clrscr();
top=NULL;
do
{
cout<<"representaion of linked stack operation \n";
cout<<"1-push a new element(s) \n";
cout<<"2-pop an element \n";
cout<<"3-display the content of the linked stack\n";
cout<<"4-exit\n";
cout<<"select your choice\n";
cin>>choice;
```

```
switch(choice)
{
case(1):{cout<<"how many element you like to enter\n";
        cin>>m;
        for(i=0;i<m;i++)
        push();
        break;
    }

case(2):{cout<<"how many element you want to delete\n";
        cin>>l;
        for(i=0;i<l;i++)
        pop();
        break;
    }
```



```

case(3):{ if(next==NULL(
            cout<<"error....linked stack is empty\n";
        else {cout<<"the content of the linked stack is\ n";
                q=top;
                while(q!=NULL(
                    {cout<<q->data<<"\n";
                    q=q->next; }
                break;
            }
        }

}

while(choice!=4);
}

```

REFERENCES

- : Introduction to Algorithms, 3rd Edition by *Thomas H. Cormen*, *Charles E. Leiserson*, *Ronald L. Rivest*, *Clifford Stein*
- Introduction to Algorithms, 3rd Edition by *Thomas H. Cormen*, *Charles E. Leiserson*, *Ronald L. Rivest*, *Clifford Stein*
- Elements of Programming Interviews in Java: The Insiders' Guide, by *Adnan Aziz*, *Tsung-Hsien Lee*, *Amit Prakash*
- <https://github.com/careermonk/DataStructuresAndAlgorithmsMadeEasy>