**Assignment No:- 1.6**

**Title:-Implementation of program based on stack using linear link list.**

#include<iostream.h>

#include<conio.h>

class NODE

{

public:

int data;

NODE \*next;

};

class STACK\_

{

private:

NODE \*top;

public:

void PUSH(int ele);

int POP();

void LIST\_ALL();

STACK\_();

};

STACK\_::STACK\_()

{

top=NULL;

}

void STACK\_::PUSH(int ele)

{

NODE \*NEW=new NODE();

NEW->data=ele;

NEW->next=NULL;

NEW->next=top;

top=NEW;

}

int STACK\_::POP()

{

if(top==NULL)

{

cout<<"\n Stack is Empty";

return NULL;

}

else

{

int ele=top->data;

NODE \*TEMP=top;

top=top->next;

delete TEMP;

return ele;

}

}

void STACK\_::LIST\_ALL()

{

if(top==NULL)

{

cout<<"\n Stack is Empty";

}

else

{

NODE \*ptr;

ptr=top;

cout<<endl<<" ";

while(ptr!=NULL)

{

cout<<ptr->data<<" ";

ptr=ptr->next;

}

}

}

void MENU()

{

int opt,ele;

STACK\_ obj;

do

{

cout<<"\n 1.PUSH";

cout<<"\n 2.POP";

cout<<"\n 3.LIST\_ALL";

cout<<"\n 4.EXIT";

cout<<"\n Enter Your Choice =>";

cin>>opt;

switch(opt)

{

case 1:

cout<<"\n ADD Element in Stack";

cin>>ele;

obj.PUSH(ele);

break;

case 2:

ele=obj.POP();

if(ele!=NULL)

cout<<endl<<ele<<"Deleted";

break;

case 3:

obj.LIST\_ALL();

break;

case 4:

return;

default:

cout<<"\n Invalid choice";

break;

}

}while(1);

}

void main()

{

clrscr();

MENU();

getch();

};