5.2. Program that Implements Stack using Linked list.

#include<conio.h> #include<stdio.h> #include<iostream> using namespace std;

class stack

{

private:

public:

int data; stack \*next; stack \*pre;

void push(int); void display(); int pop();

};

stack \*top; stack \*bottom;

void stack::push(int num)

{

if((top==NULL)&&(bottom==NULL))

{

top=bottom=new stack; top->data=num;

top->next=NULL; top->pre=NULL;

}

else

{

top->next=new stack; top->next->data=num; top->next->next=NULL; top->next->pre=top; top=top->next;

}

}

void stack::display()

{

stack \*q=bottom; if((top==NULL)&&(bottom==NULL))

{

cout<<"\nEmpty stack";

}

else

{

while(q!=NULL)

{

cout<<q->data<<" "; q=q->next;

}

}

}

int stack::pop()

{

int num; if((bottom==NULL)&&(top==NULL))

{

cout<<"\tEmpty stack"; return -1;

}

num=top->data; top=top->pre;

if( top!=NULL)

{

delete(top->next); top->next=NULL;

}

else

{

delete(bottom); bottom=NULL;

}

return num;

}

int main()

{

stack s1; char ch='y';

int num,option;

while(ch=='y')

{

cout<<"\n1.push"; cout<<"\n2.Display"; cout<<"\n3.pop"; cout<<"\n\tEnter choice: "; cin>>option;

switch(option)

{

case 1:

cout<<"enter value of element: "; cin>>num;

s1.push(num); cout<<"\tElement added.";

cout<<"\nDo you want to continue : "; break;

case 2:

cout<<"\nStack Elements: "; s1.display();

cout<<"\nDo you want to continue : "; break;

case 3:

num=s1.pop();

cout<<num << " Removed"; cout<<"\nDo you want to continue : "; break;

}

ch=getch();

}

}