**Assignment No:1.3**

**Title:Implementation of Program based on Stack.**

**Name – Mule Vishal Mukunda**

**Roll No:68**

**----------------------------------------------------------------------------------------------------------------------------------------**

#include<iostream.h>

#include<conio.h>

class stack\_70

{

private:

int \*A,s,top;

public:

stack\_70(int);

void PUSH(int ele);

int POP();

void LIST\_ALL();

};

stack\_70:: stack\_70(int par)

{

s=par; top=0;

A=new int[s+1];

}

void stack\_70::PUSH(int ele)

{

if(top==s)

{

cout<<"\n stack is Full";

return;

}

top=top+1;

A[top]=ele;

}

int stack\_70::POP()

{

if(top==0)

{

cout<<"\nstack is empty";

return NULL;

}

int ele=A[top];

top=top-1;

return(ele);

}

void stack\_70::LIST\_ALL()

{

if(top==0)

{

cout<<"\n stack is empty";

else

cout<<"\element in stack are";

for(int i=top;i>=1;i--)

{

cout<<A[i]<<" ";

}

}

void MENU()

{

int opt,ele,size;

cout<<"\n size of stack:";

cin>>size;

stack\_70 obj(size);

do

{

cout<<"\n1: PUSH";

cout<<"\n2: POP";

cout<<"\n3: LIST";

cout<<"\n4: Exit";

cout<<"\n enter your choice:";

cin>>opt;

switch(opt)

{

case 1:

cout<<"\n enter element to add:";

cin>>ele;

obj.PUSH(ele);

break;

case 2:

int ele=obj.POP();

cout<<endl<<ele<<"\n is deleted:";

break;

case 3:

obj.LIST\_ALL();

break;

case 4:

return;

default:

cout<<"\n INVALID OPTION:";

}

}while(1);

}

void main()

{

int ele;

clrscr();

MENU();

getch();

}