***//Assignment-7 Data Structures Lab***

***//A program to perform Stack operations using Array implementation***

#include<iostream>

using namespace std;

template<class T>

class Stack

{

T \*arr;

int Top,capacity;

public:

Stack(int s);

void push(T x);

T pop();

void peak();

};

template<class T>

Stack<T>::Stack (int s)

{

arr=new T[s];

capacity=s;

Top=-1;

}

template<class T>

void Stack <T>::push(T x)

{

if(Top==capacity-1)

{

cout<<"\n Stack is Overflow";

}

else

{

arr[++ Top]=x;

cout<<"\n "<<x<<" pushed";

}

}

template<class T>

T Stack<T>::pop()

{

if(Top==-1)

{

cout<<"\n Stack is Underflow";

}

else

{

T x=arr[Top--];

cout<<"\n "<<x<<" removed";

}

}

template<class T>

void Stack<T>::peak()

{

if(Top==-1)

{

cout<<"\n Stack is Underflow";

}

else

{

int i;

i=Top;

cout <<"\n The elements of the stack are: ";

for(;i>=0;i--)

{

cout<<arr[i]<< "\t" ;

}

}

}

int main()

{

Stack<int>s1(5);

int x,op;

while(1)

{

cout<<"\n\n 1.Push \n 2.Pop \n 3.Peak \n 4.Exit";

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

cin>>op;

switch(op)

{

case 1:

cout<<"\n Enter element: ";

cin>>x;

s1.push(x);

break;

case 2:

s1.pop();

break;

case 3:

s1.peak();

break;

case 4:

exit(0);

}

}

return(0);

}

**OUTPUT:**

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 2

Stack is Underflow

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 3

Stack is Underflow

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 1

Enter element: 10

10 pushed

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 1

Enter element: 20

20 pushed

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 1

Enter element: 30

30 pushed

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 1

Enter element: 40

40 pushed

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 1

Enter element: 50

50 pushed

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 1

Enter element: 60

Stack is Overflow

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 3

The elements of the stack are: 50 40 30 20 10

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 2

50 removed

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 2

40 removed

1.Push

2.Pop

3.Peak

4.Exit

Enter your choice: 3

The elements of the stack are: 30 20 10