Multi Stack

Problem Code	ds54b_quiz2_multi
Running Time Limit	1 sec
Memory Limit	32 mb

Introduction

The Stack data structure normally has three important functions: push, pop and top. In this problem, you are required to extend the Stack data structure with two more functions: **multiPush** and **multiPop**. The details of these functions are as follows

- void multiPush(Stack s, int n, DType *a) This function pushes all items in the array **a** into Stack **s**, the order of pushing starts from **a**[**n**-1] and goes down to **a**[**0**].
- DType multiPop(Stack s, int n) This function pops n items out of the Stack s. This function must return the first item popped out of the stack. It is guaranteed that n is at least 1. If n is more than the size of the Stack, it will simply pop all items from the stack. If the stack is empty, this function does nothing.

There is a starting code for this data structure at the end of this file. This starting code simulates usage of stack according to command from the input. There is one command per a line of input. There are 4 commands: push, pop, multiPush, and multiPop. The command is represented by a number from 0 to 3, respectively and it is followed by the required argument. The Stack starts as empty.

Task

Fill in the implementation of function **multipush** and **multipop** in the code below. **DO NOT modify anything in the main function.**

Input

The first line of input contains an integer \mathbf{k} , indicating the number of command. It is followed by \mathbf{k} line, each describe the command of the stack.

Output

The output contains multiple lines; each line is the result of the operation of the input. For push and multiPush, the output is the top of the stack. For pop and multiPop, the output is the return value of the function. The last line of the output is the result of printStack after all command is processed.

Example

Ex1

Input	Output
4	top: 1.0
0 1.0	top: 2.0
0 2.0	top: 3.0
0 3.0	рор: 3.0
1	stack:[1.00 2.00]

Ex2

Input	Output
3	top: 1.0
2 4 1.0 2.0 3.0 4.0	top: 10.0
0 10.0	multiPop: 10.0
3 2	stack:[4.00 3.00 2.00]

Ex3

Input	Output
3	top: 1.0
2 4 1.0 2.0 3.0 4.0	top: 10.0
0 10.0	multiPop: 10.0
3 10	<pre>stack:[]</pre>