

Query Stack

Assignment 1

Data Structures

Algorithms

Due date: 10 January, 2020

Problem Statement: There exists a stack S . You have to perform Q queries, where each query can be any one of the following operations:

- $a x$: You have to add element x to the right end of S
- t : You have to pop element from right end of S
- e : This is a new type of query. When asked to execute this operation take the left half of the stack (if there are an odd number of elements in the stack, left half is $\text{floor}(\text{total number of elements} / 2)$) and put them one by one to the right of the remaining ones.

Example: Stack is $\{1, 2, 3, 4, 5\}$. The left half of the stack is as $\{1, 2\}$ and the right half is $\{3, 4, 5\}$. We take 1 from the left half and put it to the right of the remaining elements resulting in $\{2, 3, 4, 5, 1\}$. Then we take 2 from the left half and put it to the right of the remaining elements resulting in $\{3, 4, 5, 1, 2\}$. Therefore, after 'e' operation, stack becomes $\{3, 4, 5, 1, 2\}$.

After performing all the Q queries, output a line containing N , the number of elements in the stack S left finally. The next line contains N space separated integers, the contents of the stack.

Note

This has to be in order from left end of S to the right end of S .

Input

- First line contains a single integer, Q denoting the number of queries to be performed.
- Q lines follow, where each line is of one of the following forms:
 - $a x$
 - t
 - e

where symbols have meanings as discussed in statement.

Output

- First line contains a single integer, N denoting the number of elements left in the stack after execution of Q queries.
- Second line contains N space separated integers, the contents of the stack from left to right end of the stack.

Constraints

$$1 \leq Q \leq 10^6$$

$$1 \leq x \leq 10^9$$

It is guaranteed that there exists at least one element in the stack before execution of the 't' operation.

Sample Test Case

Input	Output
8	5
a 1	4 3 5 1 2
a 2	
a 4	
a 3	
a 5	
a 8	
t	
e	