Integer Lists

Ada is developing a new programming language to work with lists of integers. The language has two built-in functions: `R` (reverse) and `D` (drop).

Function `R` reverses its input list, and `D` drops the first element of its input and returns the rest, or gives an error in case its input is an empty list. The language also allows for the composition of functions, such that the function "AB" is the function that first applies A to its input and then B to the resulting list. For example, `RDD` is a function that reverses a list and then drops the first two elements.

Ada wants you to develop an interpreted for her language. Given a program and its input, the interpreter must return the output or "`error`" in case `D` is applied to an empty list. Lists are represented as the character "`[`" followed by a comma-separated list of integers followed by the character "`]`". Notice that the input and output lists can be quite long.

Input

The first line of the input contains the number of test cases $1 \leq T \leq 10$. After that, for each of the T test cases there is one line with a string P of length at least 1 and at most 100000 representing the input program and therefore consisting of the characters "`R`" and "`D`". Following that, one line contains an the number n ($0 \leq n \leq 100000$) of elements in the input list, and another line contains the input list in the form form $[x_1,x_2,\ldots,x_n]$, with $1 \leq x_i \leq 100$.

Output

For each test case, one line containing the output list formatted as the input list, or the string "`error`" in case of error.

Examples

Sample input 1

```
4
RDD
4
[1,2,3,4]
DD
1
[42]
RRD
6
[1,1,2,3,5,8]
D
0
[]
```

Sample output 1

```
[2,1]
error
[1,2,3,5,8]
error
```

Limits

Time limit is 1 second.
Memory limit is 256 megabytes.