WC '99 P1 - The Matrix

Time limit: 1.0s **Memory limit:** 16M

Woburn Challenge 1999

Neo is trying to crack the code to break into the computer system of those omnipotent aliens. Whoa!! But even though he has been given the decryption algorithm by Morpheus, he still can't figure it out (remember, this is Keanu we're talking about here). So you need to help him break the code.

The system works as follows. The computer will give you a "challenge" - an order set of at most 200 characters, enclosed in round brackets (`(`, `)`). To successfully enter the system, you have to answer the challenge by permuting the order of the elements using the given. The correct permutation is obtained by applying one or more permutation matrices to the challenge string as follows: a permutation MATRIX of $[1\ 3\ 2]$ means that the 1st character of the challenge string stays 1st, the 3rd character is now 2nd and the 2nd character is now 3rd. So for example, if the challenge set is $(a\ b\ c)$ and the permutation matrix is $[1\ 3\ 2]$, the counterchallenge is $(a\ c\ b)$. Like we said, there may be more than one permutation matrix to apply in which case you apply them from right to left. So if the challenge is $(a\ b\ c)$ and the permutation matrices are $[3\ 2\ 1][1\ 3\ 2]$, the counterchallenge is $(b\ c\ a)$. And remember the wise words of Morpheus (i.e. us): "everyone will be told what the matrix is."

Input Specification

The input will consist of N, the number of elements in the set and the actual matrices / set on the following line.
Each "matrix" will contain at most 100 numbers, separated by spaces.
There will be at most 100 matrices on each line.
There will be no spaces between any of the brackets (`[`, `]`, `(`, `)`) and any characters following. N of -1 denotes end of data.
No line of input will contain more than 200 characters.

Output Specification

For each test case output the correct counterchallenge.
The output must contain one space separating each pair of set elements (there are no spaces between the elements and the brackets!)

Sample Input

```
3
[3 2 1][1 3 2](a b c)
2
[2 1][2 1](a b)
-1
```

Sample Output

```
(b c a)
(a b)
```