Problem A. Bijection

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

Consider paths on a plane from $(0, 0)$ to (n, n) consisting of unit steps to the right ("R") and upwards ("U"). It is known that the number of distinct such paths is the binomial coefficient

$$\text{choose}(2n, n) = \frac{(2n)!}{n! \cdot n!}.$$

For example, when $n = 2$, there are six such paths: "RRUU", "RURU", "RUUR", "URRU", "URUR", "UURR".

A string U is a regular bracket sequence if it is an empty string, or a string of the form "(V)", or a concatenation of two strings of the form "VW", where V and W are regular bracket sequences. Consider regular bracket sequences containing n pairs of brackets. It is known that the number of distinct such sequences is the Catalan number which can be calculated, in particular, as follows:

$$C_n = \frac{1}{n+1} \cdot \text{choose}(2n, n).$$

For example, when $n = 2$, there are two such sequences: "(())", "()()".

Construct any bijection that reflects this fact. More specifically, given a path of n steps to the right and n steps upwards, construct a regular bracket sequence containing n pairs of brackets, and additionally memorize an integer k from 0 to n inclusive. Afterwards, given the sequence and the integer k, restore the original path.

Interaction Protocol

In this problem, your solution will be run twice on each test.

During the first run, the solution encodes the path. The first line contains the word "path". The second line contains an integer n: half of the path length ($1 \le n \le 300$). The third line contains a path of $2n$ steps: n letters "R" and n letters "U" in some order.

On the first line, print any regular bracket sequence containing n "(" characters and n ")" characters. On the second line, print any integer k ($0 \le k \le n$).

During the second run, the solution restores the path. The first line contains the word "brackets". The second line contains an integer n, the same as during the first run: half of the bracket sequence length ($1 \le n \le 300$). The third line contains a regular bracket sequence containing n "(" characters and n ")" characters. The fourth line contains an integer k ($0 \le k \le n$). The sequence and the integer are the ones printed during the first run.

On the first line, print the restored initial path: n letters "R" and n letters "U" in the same order as in the input during the first run.

During each run, each line of input including the last one is terminated by a newline.

Examples

On each test, the input during the second run depends on the solution's output during the first run.

Two runs of some solution on the first test are shown below.

standard input	standard output
path 2 RRUU	(()) 0
brackets 2 (()) 0	RRUU

Two runs of some solution on the second test are shown below.

standard input	standard output
path 3 RUURRU	(())() 3
brackets 3 (())() 3	RUURRU