Problem I
Simon Says



Photo by David Amsler

In the game "Simon Says" one person plays the role of Simon, who gives instructions to everyone else playing the game. The tricky part is that if Simon begins his instruction with "Simon says" then everyone else *must* follow the instruction (or they lose the game); if Simon gives an instruction that does not begin with "Simon says" then everyone is supposed to completely ignore the instruction (or they lose the game)!

Simon tries his or her best to trick the other players into following the wrong instructions. Simon might begin by saying "Simon says touch your nose." and follow this with "Stop touching your nose." Anyone who stops touching their nose loses! The last player still remaining, who has correctly followed precisely the instructions that began with "Simon says" (and only these instructions), gets to be Simon next.

As a child, you were horrible at this game. Your older siblings were always able to trick you into following the wrong instructions. Well, you will have the last laugh: now that you are a computer programmer, you can write a computer program that can help you play the game perfectly. You only need to make sure the program is able to determine which instructions to follow and which to ignore.

Are you up to the challenge? Can you craft a computer program that *never* makes any mistakes in the game? If you can, then surely fame and glory shall come your way for being the most unstoppable player of Simon Says ever!

Input

Input starts with a line containing an integer $1 \leq N \leq 1\,000$. Each of the next N lines is one command, of length at most 100 characters. Each command is a properly-capitalized sequence of one or more words, separated by a single space between each pair of words, ending in a period. Some commands begin with "Simon says" and others may not. If a command begins with "Simon says", there will always be another space and at least one additional word after "says". No lines contain leading or trailing space.

Output

For each line that begins with precisely "Simon says", output the rest of the line. Each line that does not begin with precisely "Simon says" should be ignored.

Sample Input 1	Sample Output 1
1 Simon says smile.	smile.

Sample Input 2

```
3
Simon says raise your right hand.
Lower your right hand.
Simon says raise your left hand.
```

Sample Output 2

```
raise your right hand.
raise your left hand.
```

Sample Input 3

```
3
Raise your right hand.
Lower your right hand.
Simon says raise your left hand.
```

Sample Output 3

```
raise your left hand.
```