Programming task Reviewers

Memory limit: 20 MB
Time limit: 0.5 second
Input file: `reviewers.in`
Output: `reviewers.out`

Description

One day, the All About Science magazine's editor-in-chief identified problems with high-quality reviews of all articles submitted. In recent years, science in all fields has developed rapidly. There are many articles and, accordingly, many reviewers involved in the editorial staff. Managing all reviewers without a computerized system has become more complex.

It was decided to create an electronic system that will replace the existing card system. Each reviewer has a unique name consisting of a string of symbols {a..z, A..Z, 0..9} in length [1..100]. For greater anonymity, each reviewer has [1..50] unique pseudonyms. Numbers in the range [1..1'000'000'000] were used for pseudonyms. The same pseudonym cannot be assigned to any other active reviewer. At any time, one of the pseudonyms can be used to find out the specific reviewer. The editorial board decided that there would be no more than 10,000 reviewers at a time.

Because the recording was not accurate until now, one reviewer might have the pseudonym mentioned several times in the old card system (duplicates in the inserting command). It is possible to enter the reviewer information in several passes (there may be multiple inserting commands for the associated reviewer). The pseudonyms are not sorted in the insertion command because the old cards were not sorted before being entered into the new system.

An electronic information system must be created to provide a reviewer:

1. insertion or supplementation of the list of pseudonyms (new reviewers are added or the list of pseudonyms is supplemented to an already registered reviewer);
2. discarding (reviewer services are waived);
3. search (finding the reviewer's unique name, if any one of the pseudonyms is known).

After removing a reviewer from the system, new reviewers can again use pseudonyms associated with a removed reviewer.

Your task is to program the kernel of this system, which performs all operations very quickly (~ O (log (n))).

The input is a file that simulates real-life events or commands for the system. Each command with parameters is written in its row. Each row in a file may have many extra spaces between words. The file size is not limited. The input data is correct according to the input data format and the given restrictions.

Input and output (systems's commands)

Insert a reviewer or add pseudonyms to the list of pseudonyms

Input line format:

```
I name count key_1 ... key_count
```

- `I` is an abbreviation of Insert
- `name` is the reviewer's name

- `count` is the number of pseudonyms [1..50] assigned to the reviewer
- `key_1 ... key_count` are pseudonyms separated by spaces

Output line containing:

- the word `ok`, if the reviewer has been successfully inserted or the list of pseudonyms can be supplemented for the existing reviewer,

```
ok
```

- the word `no`, if the reviewer cannot be inserted, because some pseudonym has already been used for another reviewer in the system or the number of unique pseudonyms for the reviewer will exceed 50. In this case, no changes are made in the system!

```
no
```

Discarding a reviewer

Input line format:

```
D key
```

- `D` is an abbreviation of Delete
- `key` is one of the pseudonyms of the discarded reviewer

Output line containing:

- the word `ok`, if the reviewer has been successfully discarded

```
ok
```

- the word `no`, if the reviewer cannot be discarded because there is no reviewer with the specified pseudonym

```
no
```

Searching for a reviewer

Input line format:

```
L key
```

- `L` is an abbreviation of Look_up
- `key` is one of the pseudonyms of the searched reviewer

Output line containing:

- name of the reviewer, if the reviewer has been successfully found,

```
Name
```

- the word `no`, if the reviewer cannot be found because there is no reviewer with the specified pseudonym.

```
no
```

Example:

The content of input file `reviewers.in` :

```
I JackSmart 3 9 1009 1000009
L 1000009
I TedPumpkinhead 1 19
I PeterMeter 1 9
L 19
D 19
L 19
I JohnCritic 2 1 19
L 19
L 1
L 9
```

The content of output file `reviewers.out` :

```
ok
JackSmart
ok
no
TedPumpkinhead
ok
no
ok
JohnCritic
JohnCritic
JackSmart
```