MOD and MOD_IF

Time Limit: 5 Seconds Memory Limit: 65536 KB

ZOJ is 10 years old! For celebrating, ZOj has a new storage system. In this new storage system, data is organized as many records, and each record has several attributes. For example, there can be two records called R1 and R2 with two attributes 'age' and 'height':

==	age	height
R1	10	60
R2	30	175

Modification operation in this system is defined as two types: MOD and MOD_IF. MOD simply sets the given attribute with the given value. MOD_IF sets the given attribute with the given value only if the value of the attribute in "if" clause equals to the given value. For example, we have a list of operations:

- mod age 5
- mod age 3 if height 175

When the operations applied to the previous data example, we will get:

==	age	height
R1	5	60
R2	3	175

In the system, when an operation list is submitted, the operations in the list will be applied on the first record in the system one by one. Then the second record in the system is chosen, all the operations apply to this record one by one. So on and so force, until all the records in the system are applied.

Now we have our question here, can we remove some of the operations in the list while the remaining operations work just the same for any possible data record?

Because of some flaws, the operation removal has the following principal. An operation X in the list can be removed if and only if in this situation: after all the previous operations are applied to any possible record, this operation X won't change anything.

Input

The input consists of several test cases. For each case there are n ($1 \le n \le 100000$) lines of operations followed by an "END". "OVER" indicates the end of the whole input.

An operation can be either of format "mod ATTR VALUE" or "mod ATTR VALUE if ATTR VALUE", ATTR is a character within {'a','b','c','d','e'}, VALUE is integer in [0, 9]. Note: the two ATTR in mod_if can be same. There may be empty lines in the input.

Output

For each test case, output each operation, if the operation can be removed, output an "R" at the tail of the operation. Output a line with "END" at the end of each test case.

Sample Input

```
mod a 1
mod a 1
mod a 2
END
mod a 1 if b 2
mod a 1 if b 2
END
OVER
```

Sample Output

```
mod a 1
mod a 1 R
mod a 2
END
mod a 1 if b 2
mod a 1 if b 2 R
END
```

Author: **ZHENG, Jianqiang**