Using money to pay for goods and services usually makes life easier, but sometimes people prefer to trade items directly without any money changing hands. In order to ensure a consistent "price", traders set an exchange rate between items. The exchange rate between two items A and B is expressed as two positive integers m and n, and says that m of item A is worth n of item B. For example, 2 stoves might be worth 3 refrigerators. (Mathematically, 1 stove is worth 1.5 refrigerators, but since it's hard to find half a refrigerator, exchange rates are always expressed using integers.)

Your job is to write a program that, given a list of exchange rates, calculates the exchange rate between any two items.

Input

The input file contains one or more commands, followed by a line beginning with a period that signals the end of the file. Each command is on a line by itself and is either an assertion or a query. An assertion begins with an exclamation point and has the format

$$! \ m \ itema = n \ itemb$$

where *itema* and *itemb* are distinct item names and m and n are both positive integers less than 100. This command says that m of *itema* are worth n of *itemb*. A query begins with a question mark, is of the form

$$? \ itema = itemb$$

and asks for the exchange rate between *itema* and *itemb*, where *itema* and *itemb* are distinct items that have both appeared in previous assertions (although not necessarily the same assertion).

Output

For each query, output the exchange rate between *itema* and *itemb* based on all the assertions made up to that point. Exchange rates must be in integers and must be reduced to lowest terms. If no exchange rate can be determined at that point, use question marks instead of integers. Format all output exactly as shown in the example.

Note:

- Item names will have length at most 20 and will contain only lowercase letters.
- Only the singular form of an item name will be used (no plurals).
- There will be at most 60 distinct items.
- There will be at most one assertion for any pair of distinct items.
- There will be no contradictory assertions. For example, "2 pig = 1 cow", "2 cow = 1 horse", and "2 horse = 3 pig" are contradictory.
- Assertions are not necessarily in lowest terms, but output must be.
- Although assertions use numbers less than 100, queries may result in larger numbers that will not exceed 10000 *when reduced to lowest terms.*

Sample Input

```
! 6 shirt = 15 sock
! 47 underwear = 9 pant
? sock = shirt
? shirt = pant
! 2 sock = 1 underwear
? pant = shirt
.
```

Sample Output

```
5 sock = 2 shirt
? shirt = ? pant
45 pant = 188 shirt
```