

Dear Programmers:

I can imagine how surprised you are when you receive this strange etter. Well, be patient, I am going to talk about some really exciting things that you must be interested in.

First of all, I have to congratulate everyone of you - the gifted programmers, for you are so lucky to have a God-given chance to be my inheritor - a new Deity of Stars. However, you know that it is difficult for an ordinary person to become a deity, so only the best one among you can finally be chosen. Therefore, you are facing a hard test for estimating your abilities. Try your best to compete!

The Deity of Stars, of course, is the unique dominator of all stars in the whole universe, which means controlling the stars' tracks and ensuring that the stars move in their own orbit are his/her most important responsibilities. What? You will be sick in managing incalculable stars? Don't worry, it will be a piece of cake once you have a wonderful tool - a special kind of programming language. That is why I am going to choose my inheritor from you.

So in your test, you will be given a special program and the initial positions of some stars (In deities' eyes, the stars are so small that they can be regarded as points), you need to figure out the new positions of these stars after the program is executed.

Considering all of you are green hands, I will just give you a program in a simple version during the test, which contains only several kinds of instructions listed below:

1. **translate** $tx\ ty\ tz$ Everything in (x, y, z) must be moved to $(x + tx, y + ty, z + tz)$
2. **scale** $a\ b\ c$ Everything in (x, y, z) will be moved to (ax, by, cz)
3. **rotate** $a\ b\ c\ d$ It will make everything rotate. The rotation axis is the straight line from $(0, 0, 0)$ to (a, b, c) , and the angle of rotation is d (measured in degrees). If you stand at (a, b, c) and look at $(0, 0, 0)$, you will see that the rotation is counterclockwise.
4. **repeat** k The instructions between a "repeat" instruction and an "end" instruction matched with it (will be explained later) will be executed for k times. The integer k is non-negative and a 32-bit signed integer is sufficient to deal with it.
5. **end** If there are some unmatched "repeat" instructions, the "end" instruction will be matched with the nearest unmatched "repeat" instruction before it; otherwise, it indicates the end of the program. Please note that a repeat-end pair may include other repeat-end pairs.

Now the test is coming. Are you ready?

Good Luck,
Zi Wei, Emperor of North Polaris

Input

The input contains no more than 20.

Each test case begins with one integer n ($1 \leq n \leq 1000$), indicating the number of the given points. Then there is a correct program without any extra blanks or redundant characters and it contains less than 100 lines. Each line contains only one instruction. Then n lines follow, each containing three numbers which indicate the coordinates of a point in 3D universe. All numbers except n and k are floats and no more than 1000 in absolute value.

Two consecutive cases are separated by a blank line.

The input ends with $n = 0$.

Output

For each test case, print n lines, each line contains three float numbers indicating the new position of a point. Please round the results to two digits after decimal point. You should print the n points in the same order as in input.

Print a blank line **after** each test case.

Sample Input

```
2
rotate 1.0 1.0 1.0 90.0
translate 2.0 2.0 2.0
end
1.0 1.0 1.0
1.0 0.0 0.0

3
repeat 100
translate 2.7 -0.2 1.1
translate -2.6 0.0 -1.0
end
scale 1.0 0.0 0.5
end
0.5 2.7 1.1
0.22 0 7.0
1.2 3.4 5.6
0
```

Sample Output

```
3.00 3.00 3.00
2.33 2.91 1.76

10.50 0.00 5.55
10.22 0.00 8.50
11.20 0.00 7.80
```