5 Point Defense

During a battle, a computer virus attack from enemy spacecraft disabled your anti-missile point defense programs and destroyed the backups. You can feel smug that your last readiness report included a note that there should be offline backups of all critical programs on board. However, before you can feel truly self-superior, you have to survive the afternoon, and it is becoming increasingly less likely as the enemy ships seem to have other ideas. Under fire, you must re-implement the important features of the point defense programs.

The program must determine if each missile will hit the ship within a tolerance of 0.000000001 units. Then in what order the enemy missiles will hit your ship. Given this list, it must produce firing solutions for your point defense guns in the order that the missiles will impact.

The input will consist of a set of missile position points and velocity vectors, relative to your own position and velocity vector.

The format of each line of input will be:

$$(x, y, z)\ [v_x, v_y, v_z]$$

The output will consist of a unit vector (a vector with magnitude 1) of the direction to fire the point defense gun for each missile in the order the enemy missiles will hit the ship (one per line):

$$[v_x, v_y, v_z]$$

You may assume that no two missiles will hit at exactly the same instant. All output should be rounded and displayed with three digits of precision after the decimal point.

Note: The ¶ symbol in the examples below represents a newline character.

Sample Input 1

```
(0.0,0.0,10.0) [0.0,0.0,-1.0]¶
(0.0,0.0,-9.0) [0.0,0.0,1.0]¶
(0.0,5.0,0.0) [0.0,1.0,0.0]¶
```

Sample Output 1

```
[0.000,0.000,-1.000]¶
[0.000,0.000,1.000]¶
```

Sample Input 2

```
(10.0,10.0,10.0) [-5.0,-5.0,-5.0]¶
(0.0,0.0,10.0) [0.0,0.0,-10.0]¶
```

Sample Output 2

```
[0.000,0.000,1.000]¶
[0.577,0.577,0.577]¶
```