6635 Shoot

At the year of 8192, the war between Evil Army and Galaxy Army broken out. Unfortunately,Evil Army had conquered half the galaxy in just one year. To prevent the situation of the war from getting worse, Levi, the general of Galaxy Elite Army, was ordered by his superior to attack the enemy's power bases.

Levi was born with the remarkable ability of counter-surveillance, it was just a piece of cake for him to reach Evil Army's power bases. Each power base can be represented as a triangle in 3D-Cartesian coordinate system. The only weapon Levi had was a laser cannon which can shoot in both two directions simultaneously. To avoid being caught by enemy, Levi can only place the laser cannon somewhere on a segment from S to T. Unfortunately, there was something wrong with the laser cannon, Levi can't adjust its shooting angle, so the shooting direction was fixed.

Since Levi didn't have any time to find the best place to shoot the laser, he decided to select a point on the segment randomly to place the cannon. If the laser touched the base (even the boundary), the base will be destroyed. Your task is to calculate the expected number of the destroyed bases in just one shoot.

It is recommended to see the sample input to understand the problem statement more clearly.

Input

There are several test cases, please process till EOF.

For each test case, the first line is an integer N ($1 \leq N \leq 100000$), the number of enemy's power bases. Each of the next three lines contains 3 integers, x, y, z, denoting the coordinates of S, T, and the fixed shooting direction. The last N lines contains 9 integers, x_1, y_1, z_1, x_2, y_2, z_2, x_3, y_3, z_3, denoting the coordinates of the three vertices of enemy's power base. It is guaranteed that all the triangles will not degenerate.

And the absolute value of all numbers except N will not exceed 1000.

Output

For each test case, print the expected number of destroyed power bases.

Any answer within an absolute error less than or equal to 10^{-6} would be accepted.

Sample Input

```
2
0 0 0
2 0 0
0 0 1
-1 0 1 1 0 1 -1 0 2
1 1 -1 1 -1 -1 2 0 -1
```

Sample Output

```
1.00000000
```