

Jack Skellington kidnapped Sandy Claws so he can replace him on Christmas! The monsters of Halloween Town are helping Jack make toys for all the children.

One of these toys is a special kind of gyroscopic exercise tool. It consists of n balls that levitate around the center of the toy. To play with it, a child chooses some axis passing through the center and begins to rotate all the balls around it simultaneously. The effort needed to perform the exercise is proportional to the sum of squared distances from every ball to the chosen axis. If the needed effort for this axis is not less than for any other one, we call it *The Axis of Awesome*. Now Jack wants to improve these gyroscopic toys by adding zero or more balls to each toy so that every possible axis of play was an Axis of Awesome.

You are given the blueprints for t gyroscopic toys, where each toy is described as a set of n three-dimensional (x, y, z) coordinates denoting the locations of the toy's balls. For each toy, find the minimum number of balls Jack must add so that any possible axis would be called Axis of Awesome. If no amount of additional balls makes this possible, print `-1` instead.

Note: The center of each toy is always located at point $(0, 0, 0)$.

Input Format

The first line contains an integer, t , denoting the number of gyroscopic toys. The subsequent lines describe each toy in the following format:

1. The first line contains an integer, n , denoting the number of balls in the toy.
2. Each of the n subsequent lines contain **3** space-separated integers describing the respective x , y , and z coordinates of one of the toy's balls.

Constraints

- $1 \leq t \leq 5000$
- $1 \leq n \leq 100$
- $-50 \leq x, y, z \leq 50$
- Different balls *may* have the same coordinates.

Output Format

For each toy, print a single integer on a new line denoting the minimum number of balls Jack must add to the toy so that the effort to play with it is always maximal; if this is not possible, print `-1` instead.

Sample Input 0

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

1 0 0

0 1 0

Sample Output 0

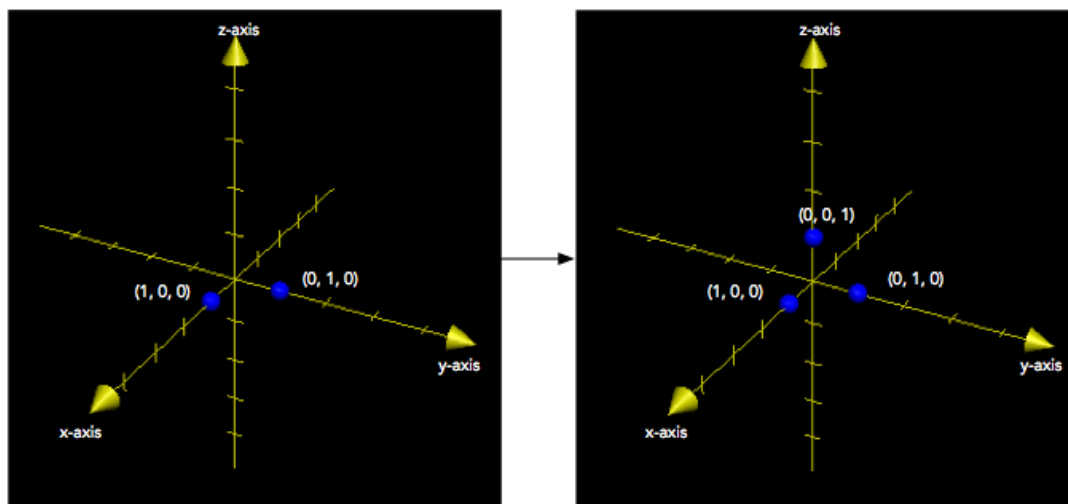
0

1

Explanation 0

Jack wants to improve the following $t = 2$ toys:

1. For the first toy, you don't need to add any balls because the effort is always **0** and doesn't depend on the chosen axis. Because we didn't need to add any balls, we print **0** on a new line.
2. For the second toy, we can add a ball at point **(0,0,1)** so that the total squared distance from the updated set of balls to any axis is equal to **2**:



Because we only needed to add one ball, we print **1** on a new line.