Plan B for next SWERC

Background

As we all know, teams participating in ICPC or in any of the previous regionals receive one balloon for each problem solved. At the end of the contest, the team with more balloons wins.

As part of the selection process, the candidate members of our team for the next Southwestern Europe Regional Contest need to prove that they know how to handle our new secret weapon: an USB controlled laser cannon. In the highly unlikely case that our team got less balloons than some other team, they would use the laser cannon to burst other teams' balloons.

The Problem

We will assume that the contest takes place in a two dimensional room and that the only objects in the room are the target balloons and the laser cannon. Further, our laser has infinite range and the balloons are perfect circles. The cannon is in the lower left corner of the room (which has coordinates (0, 0)). Hence, to aim the cannon, the user only has to select the proper elevation angle.

Unfortunately, the laser cannon is a cheap model from ACME that can only fire at elevations from 0 sexagesimal degrees up to 90, in increments of 10 degrees. Also, it sometimes gets stuck when lowering the muzzle, hence if several shots are to be fired, it is important to fire them in order from lowest to higher elevation angle.

You have to write a program to calculate at which elevations the cannon has to be fired and how many balloons will be burst, given the size and coordinates of all the target balloons.

The Input

The input format is as follows:

An integer in a single line which says the number of problems to solve. Then, for each problem:

- An integer in a line of itself denoting the number of target balloons.
- One line for each balloon, with three integers giving the coordinates *(x, y)* of each balloon and its diameter (all distances are given in centimeters). No balloon is further than 15 meters from the cannon and all balloons are smaller than 1 meter in diameter.
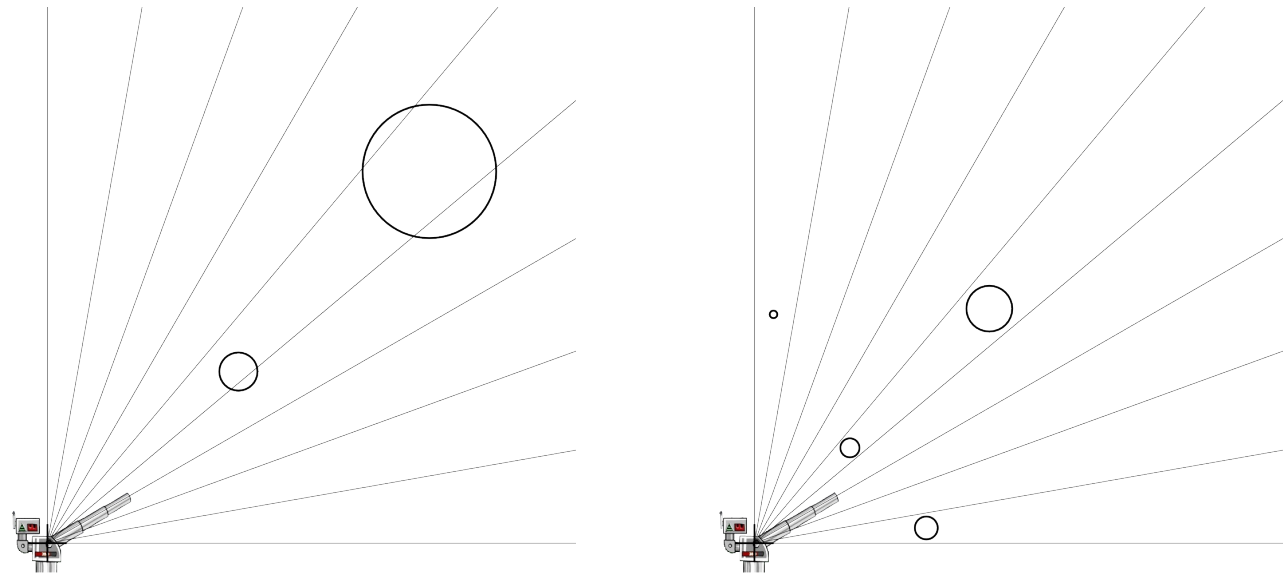
The Output

The output for each problem consists of one line for each shoot of the form "Fire laser at *d* degrees.", where *d* is the elevation for the shot, followed by a line indicating how many balloons have actually been burst in the form "1 burst balloon." when only one balloon has been burst, "No balloon burst." if no balloon has been harmed and "*b* burst balloons." if *b* balloons have been hit and *b > 1*.

Sample Input

```
2
2
100 90 20
200 195 70
4
123 123 25
90 8 12
50 50 10
10 120 4
```

The following figures are diagrams showing the positions of the balloons in the two sample problems and all the possible firing elevations:

Sample Output

```
Fire laser at 40 degrees.
2 burst balloons.
No balloon burst.
```