ETH Zurich
Institut für Theoretische Informatik
Prof. Dr. Angelika Steger
Prof. Dr. Emo Welzl
Prof. Dr. Peter Widmayer

HS 2014
Exercise Sheet Problem of the Week VIII

Algorithms Lab

Exercise 1 – *Collisions*

For the whole country of Sealand as well as for a considerable sea area around it, air surveillance is handled by the FlyGuide unit at Lagoontown airport. During a recent visit to the neighboring country of Beerland including an extensive tasting of local specialties, Mr. Noah Diah—Sealand's Minister of Tourism and Transportation—bragged about the advanced collision detection system that FlyGuide uses to warn about potential collisions between airplanes: The system can handle several thousands of airplanes simultaneously, though admittedly in practice rarely more than ten planes cruise above Sealand at any given moment. Unfortunately, this statement has drawn some attention among the Beerland delegation, in particular from their minister of Science, Brewing and Technology, Ms. Donna Tbileef, who spontaneously scheduled a visit at FlyGuide in less than two weeks from now. Even more unfortunately, FlyGuide has never even thought about a system to handle so many planes simultaneously. Nevertheless, the personal assistant of Mr. Diah has hinted very clearly at a rich reservoir of unpleasant consequences in the unlikely event that the system should not be there for Ms. Tbileef to inspect.

Please help the people at FlyGuide to set up a rudimentary large-scale collision detection system. The goal is to process a set of planes and warn about the ones that are too close to each other. For a first version of the system, we assume that all planes fly at the same altitude.

Input The first line of the input contains the number $t \leq 60$ of test cases. Each of the following t test cases is described as follows. It begins with a line containing one integer n ($3 \leq n \leq 2^{20}$), denoting the number of planes. Then follows a line containing one integer d ($1 \leq d < 2^{25}$), denoting the intended minimal distance between any two planes. The next n lines describe the current position of all planes. Each position is defined by two integer coordinates `x y`, separated by a space and so that $|x|, |y| < 2^{25}$. Coordinates and distances are measured in meters. You may assume that the input points in each test case are unique, that is, no two planes are at the same position.

Hint: Please use the datatype `int` to read the coordinates from `std::cin`.

Output For each input, the output appears on a single and separate line, which gives the number of planes that are in danger. A plane is *in danger*, if and only if there is some other plane in distance less than d.

Sample Input

```
2
3
5000
0 0
-100 -100
3535 3535
3
5000
0 0
3536 3536
-3536 3536
```

Sample Output

```
3
0
```