

Dauids Greed

The King David it's a particularly miser character. He have all the gold in his kingdom, and never lends a single coin to anyone. Years passed, and the streets in his kingdom, made by his father, King Enrique, started to deteriorate. The villagers couldn't traffic with their carts without dropping some of their content.

This problem was affecting the merchants, because they were losing food and other stuff that they sell. So they decided to go to the castle and block the entrance, so the king couldn't get out until he promised that he would restore the streets.

The King, tired by the annoyance of the people, decided to promess to restore the streets of his kingdom, but because he's too greedy, he will restore only the ones that are important for the city, let's call them "vital streets".

The vital streets connect a commercial point A to a commercial point B in the city such that there will never exist 2 ways to go from point A to point B. This is given by some coordinates X and Y, you can assume the commercial point will be a single point in a space.

The original streets, built by King Enrique, connects every point with the other points, so point A it's connected with point B, C, D, and so on. So restore every street would be too expensive, that's why King David will restore only the "vital streets", but he is not really good at this type of calculations, so he need you to write a program that help him out.

The cost of restoring a street is given by its length and some value P, that tell how much cost to restore a single unit of distance. By example, if $P = 3$, and the distance of points A and B it's 3.5, then the total cost of restoring that street would be $3.5 * 3$. Note: you have to round up the total cost of the street.

Input

The first line of input will consist of an integer T, denoting the number of test cases. Every test case will have by first line 2 integers, N and P, denoting the number of points and the unitary cost. Then N lines follow, every of them having 2 numbers X_j and Y_j , denoting the coordinates in the cartesian plane of every commerciant points.

Output

For each test case output a single line containing this: "Scenario #i: j", where i is the number of the test case, starting in one, and j it's the minimum cost of restoring the streets, such as the restored streets let the villagers to go from one point to another point. Since this number can be quite large, output $j \bmod 10^9 + 7$.

Example

Input:

```
2
4 1
35 46
29 13
44 0
27 17
3 18
18 0
11 17
2 31
```

Output:

Scenario #1: 56
Scenario #2: 631

Constraints:

$$1 \leq N \leq 1,000$$

$$-1,000 \leq X_i, Y_i \leq 1,000$$

$$1 \leq P \leq 1,000$$