

# Late Summer Searching

Summer is late! Summer is late! For a very important date! The VPL programming contest is about to start and Summer is nowhere to be found. Without Summer, you cannot have a contest based on seasons, so you have to find her quickly!

A system of antennas is placed on the top of the tallest buildings. You can use them to try to detect Summer's position, so you can finally go and get her. However, the antennas are very sensitive to noise, so unless every antenna is visible from every other antenna, the system will not work. Every antenna has an interference radius  $R$ , that will block the visibility between two other antennas if their signal passes through it.

Even if the system works, using it is very energy-consuming and quite expensive. Given the positions of the antennas, the cost of using the system is equal to the minimal possible area that encloses every antenna (along with their lines of sight with every other antenna). For simplicity, you can suppose that the physical radius of each antenna is negligible.

You are located in the VPL headquarters building (which has its own antenna), where every other building with an antenna has a position relative to yours. Knowing these positions, and the interference radius for the antennas, your job is to decide whether the system can work or not. And if it can, what will be the cost (in terms of the minimal area covered).

## INPUT DETAILS:

The first line of the input will contain a single integer  $T$ , which is the number of test cases that follow. Each test case will begin with a line containing two integers  $N$  (the number of antennas) and  $R$  (Radius of interference of the antennas).  $N-1$  lines follow, each one containing two integers  $X_i$  and  $Y_i$ , being the relative position of the  $i$ -th building with respect to VPL headquarters (which is assumed to be placed at position  $(0, 0)$ ).

## OUTPUT DETAILS:

The output of your program should consist of  $T$  lines, one for each test case. For each of these test cases you should output a single number, representing the minimum area covered by the system, (Rounded to two decimal places). If the system cannot be used, you should output only the text "NO CONTEST" (quotes for clarity).

INPUT	OUTPUT
2	4.00
4 1	NO CONTEST
2 0	
2 2	
0 2	
5 1	
2 0	
2 2	
0 2	
1 1	

**GENERAL CONSTRAINTS:**

$$1 \leq T \leq 100$$

$$1 \leq R \leq 10$$

$$-500 \leq X_i, Y_i \leq 500$$

**CONSTRAINTS - SUBTASK 1 (20%):**

$$N = 3$$

**CONSTRAINTS - SUBTASK 2 (80%):**

$$3 \leq N \leq 100$$