

Aliens

Aliens visited our planet with an obvious intention to find some new species for their space zoo. After entering Earth's orbit, they positioned themselves over the town of Belgrade, having detected some life-form activity on the ground. As they approached the surface, they saw a group of half-intelligent beings. Those creatures were actually competitors of the Balkan Olympiad in Informatics who were enjoying the excursion after intense contest. Aliens want to abduct all n ($2 \leq n \leq 100000$) competitors since they are very compassionate, and don't want their creatures to feel lonely in the space zoo.

Aliens use tractor beam to take their prey. Tractor beam works in the following way: it projects a circle-shaped beam from the spacecraft to the ground vertically beneath it, and all beings that are found in that circle or on its boundary are taken. Projecting the tractor beam needs a certain amount of energy to be spent. As the radius of the tractor beam (radius of the circle on the ground) increases, more and more energy is required. Although extremely intelligent, aliens are much more advanced in social sciences than in programming. That's why they are asking you to help them find the position of their spacecraft so that the energy required to take all of the n competitors is minimal.

Help our alien brothers! Write a program that will find the required minimal radius of tractor beam that contains all n competitors and the optimal spacecraft location - which is the same as the center of the circle on the ground.

Input

First line of input contains one integer $c \leq 20$ - number of test cases. Each test case begins with number n ($2 \leq n \leq 100000$). Then n lines follow and i -th of them contains two real numbers x_i and y_i ($-10000.0 \leq x_i, y_i \leq 10000.0$) representing coordinates of the i -th competitor.

Output

For each test case output radius of the tractor beam and coordinates of the spacecraft. Numbers should be rounded to two decimal places.

Example

Input:

```
1
6
8.0 9.0
4.0 7.5
1.0 2.0
5.1 8.7
9.0 2.0
4.5 1.0
```

Output:

```
5.00
5.00 5.00
```

Warning: large Input/Output data, be careful with certain languages