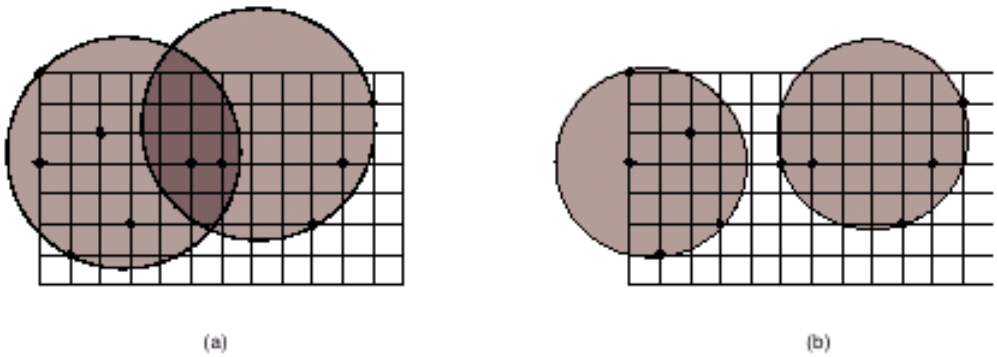
A new telephony company intends to offer services of residential telephone in your city. The phones will be residential, but the company will use mobile tecnology, with transmitter towers, to avoid the expenses of building a cable network throughout the city.

The power of the equipment installed in a tower defines the tower's *cover radius* (which, in turn, defines the cover area of the tower, since the city is perfeclty flat). The cost of the equipment installed in each tower is directly proportional to its power, and therefore to its cover radius.

The company decided that exactly two towers will be used. The same type of equipment will be installed in both towers; that is, both towers will have the same cover radius. Since the company wants to offer its services to all homes in the city, the cover radius of both towers, together, must cover all homes. Additionally, to minimize the cost, the cover radius of the towers must be the minimum possible. The figure below shows two possible configurations of cover radius for the two towers in a city with ten homes. Both (a) and (b) cover all homes, but (b) is the one that uses the minimum possible cover radius.



Given the position of each home in the city, you must write a program to determine the smallest cover radius for the towers, so that all homes in the city are covered.

Input

The input contains several test cases. Each test case is composed by two lines. The first line of a test case contains an integer N, the number of homes in the city ($3 \leq N \leq 40$). Each of the following N lines contains two integers X and Y, separated by a single space ($0 \leq X \leq 10^4$ e $0 \leq Y \leq 10^4$), representing the position of a home. Each home has a different position.

The end of input is indicated by a line containing only one zero.

Output

For each test case in the input your program must print a single line, containing a real number, written with a precision of two decimal places, indicating the smallest cover radius to be used in both towers.

Sample Input

```
3
0 0
1 0
0 4
10
0 0
0 3
1 6
2 2
3 5
5 3
6 3
9 5
10 5
11 3
0
```

Sample Output

```
0.50
3.05
```