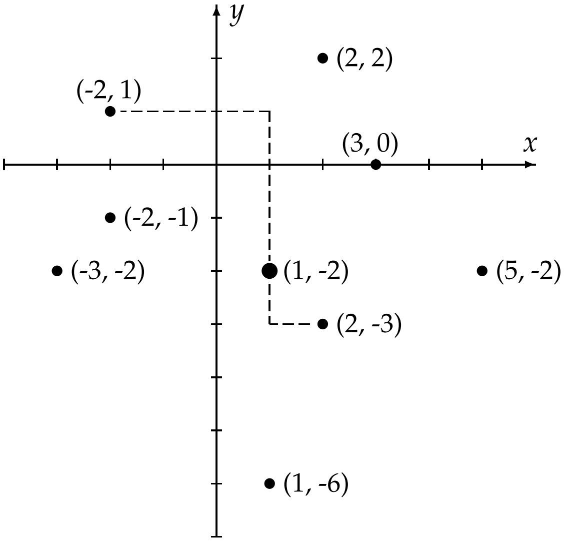
F006B. Manhattan distance **P64555_en**

The Manhattan distance between two points with integer coordinates in a plane is described as the number of unit steps towards up, down, left or right that you must do to go from a point to the other one. (It has this name because is the number of blocks that is necessary to walk in a squared neighborhood to go from a junction to another one.)



For instance, consider the point (1, -2) of the figure on the right (the biggest). The nearest point is (2, -3), that it is in a distance 2. The farthest point is (-2, 1), that it is in a distance 6. Except that (2,2), the rest of points are in a distance 4.

Your task is to write a program that, given a point (x,y) and n points $(x_1,y_1),\ldots,(x_n,y_n)$, prints these points sorted depending on their Manhattan distance to (x,y).

Input

The input consists of integer numbers, and it is formed by one line with x and y, one line with n, and one or more lines with the coordinates of the n points: $x_1,y_1,x_2,y_2,\ldots,x_n,y_n$. You can suppose $0 \le n \le 10^5$. The n points are different, and they can be in any order.

Output

Your program must print according to their distance to (x,y). If two points are at the same distance, print first the one that has the first coordinate smaller and, in the event of draw, the one that has the second coordinate smaller. Follow the format of the instance.

Observation

Your algorithm must be efficient in all the cases, because n can be huge, and because the private test data will include borderline cases, like a lot of points at the same distance.

Sample input 1

```
1 -2
8
-3 -2
-2 -1
-2  1
1 -6
2 -3
2  2
3  0
5 -2
```

Sample output 1

```
points at distance 2
2 -3
points at distance 4
-3 -2
-2 -1
1 -6
3 0
5 -2
points at distance 5
2 2
points at distance 6
-2 1
```

Sample input 2

```
2 2
13
            2 4
      1 3   2 3   3 3
0 2   1 2   2 2   3 2   4 2
      1 1   2 1   3 1
            2 0
```

Sample output 2

```
points at distance 0
2 2
points at distance 1
1 2
2 1
2 3
3 2
points at distance 2
0 2
1 1
1 3
2 0
2 4
3 1
3 3
4 2
```

Sample input 3

```
-100000000 -100000000
4
-1 -1
0 0
50000000 50000000
-87654321 87654321
```

Sample output 3

```
points at distance 199999998
-1 -1
points at distance 200000000
-87654321 87654321
0 0
points at distance 300000000
50000000 50000000
```

Sample input 4

```
2 3
0
```

Sample output 4

Problem information

Author : Professorat de P1
Translator : Carlos Molina
Generation : 2013-09-02 15:09:46

© *Jutge.org*, 2006–2013.
http://www.jutge.org