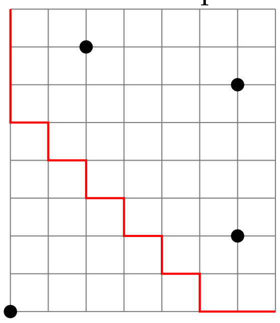
The Smoker's problem

A smoker (with a lit cigarette) wishes to walk from the northwest corner of a large gridded network of walkways to the southeast corner. Seated at benches at some of the intersection points are non-smokers. Fearing violence, or showing consideration, the smoker wants to keep as far away from the non-smokers as possible. That is, he aims to find a path whose minimum distance from a non-smoker is as large as possible (the distance between two points in the grid is the sum of their horizontal and vertical separations, and the distance between a point and a path is the minimum distance between the point and any point on the path).

In a further fit of kindness, he also wants to ensure that, subject to meeting the first criterion, the *total* distance between the path and the non-smokers should be as large as possible.

Example

In the grid below non-smokers are marked with dots, and a "best" path is drawn in red. This is justified as follows – clearly the upper left non-smoker and lower right non-smoker mean the best minimum distance is 2 (and that distance is forced for those two points). The maximum distance we could have for the upper right non-smoker is 6 since we have to cross her horizontal line somewhere. Our first two steps are forced, so the best we could do for the lower left is also 6 - but the third step must reduce the distance to either the upper right or lower left by 1. Choosing a down step, we then succeed in keeping our distance from those two points by the zig zag pattern.



Task

Write a program to find the optimum minimum distance, and also the optimum total distance for that minimum in a sequence of scenarios contained in an input file read in from `stdin`.

Input format

- Each scenario begins with a line containing the number of rows and number of horizontal walkways and the number of vertical walkways.
- Each additional line of a scenario describes the locations of the non-smokers (with the upper left at `0 0`), increasing from left to right and top to bottom.
- Each scenario is separated from the next by a blank line.

You may assume that there are at most 100 vertical and 100 horizontal walkways, and at most 500 non-smokers in a scenario.

The scenario corresponding to the example above would be:

```
9 7
0 8
2 1
6 2
6 6
```

Output format

For each scenario print (to `stdout`) a line of the following form:

```
min M, total T
```

where `M` and `T` are the largest possible minimum distance achievable and the largest possible total distance achievable for the minimum. So, the output for the scenario above should be

```
min 2, total 15
```

(3 points, Group)