Canal Navigation (prob4)

A canal connects two bodies of water. Sometimes the height of the two bodies of water that are connected are different and a series of locks are necessary to construct the canal. In this problem you will have to write a program that allows a boat to travel from one cell of a lock system in a canal to another cell of a lock system in minimal time.

To simplify the problem, assume that the locks create a rectangular arrangement of cells and each cell has the shape of a square with a lock on each of the four sides. Each cell has an elevation of water. We will assume that the boat in question always starts from the northwest corner of the lock system and needs to travel to the south east corner. In each "move", the boat may choose to go south or east, unless the cell in which it's in is southernmost or easternmost. The amount of time in minutes a move between cells takes is simply the difference in elevation in feet of the starting cell before and after the move. After the move, both cells will have the same elevation and then the corresponding lock is put back in place.

Consider the following lock system with six cells:

3	9	5
1	4	5

On our first move, if we move south, we would temporarily open the southern lock of the northwest square. This would make the water levels in both of the westernmost cells equal to 2, taking 1 minute, since the elevation of the first cell changes from 3 to 2 feet. Thus, our ensuing picture is as follows:

2	9	5
2	4	5

From here, if we move east twice, we will arrive at the desired destination in 3 minutes total, since each of the last two moves take one minute as well. Here is a complete look at each move:

3	9	5
1	4	5

→

2	9	5
2	4	5

Cost = 1

→

2	9	5
3	3	5

Cost = 1

→

2	9	5
3	4	4

Cost = 1

This sequence of moves leads to the fastest path to move from the northwest corner to the southeast corner. The other two possible paths would take 3.75 minutes and 4 minutes. These are shown below.

3	9	5
1	4	5

→

6	6	5
1	4	5

Cost = 3

→

6	5.5	5.5
1	4	5

Cost = 0.5

→

2	5.5	5.25
3	4	5.25

Cost = 0.25

3	9	5
1	4	5

→

6	6	5
1	4	5

Cost = 3

→

6	5	5
1	5	5

Cost = 1

→

6	5	5
1	5	5

Cost = 0

Input

The first line of the input file will contain a single positive integer, *n*, representing the number of lock systems to navigate. Data for each of the *n* lock systems follows.

The first line of data for each lock system will contain two positive integers, *r* ($r \leq 10$) and *c* ($c \leq 10$), representing the number of rows and columns in the grid system, respectively, separated by spaces. (The first row represents the north row of the grid and the first column represents the west column of the grid.) The following *r* lines contain *c* positive integers less than or equal to 1000, each separated by spaces, representing the elevation of each of the cells on the given row, from west to east.

Output

For each lock system, output a single line with the following format

```
Canal k: X
```

where k is the number of the lock system, starting with 1, and X is the number of minutes representing the minimum time to move in the lock system from the northwest corner to the southeast corner with south and east moves only, rounded to three decimal places.

Sample Input

```
2
2 3
3 9 5
1 4 5
2 2
1 11
2 3
```

Sample Output (corresponding to sample input)

```
Canal 1: 3.000
Canal 2: 1.250
```