**Problem 3. Robot(robot)**

The software company in which Pesho is working, ordered Pesho to develop the software for their new drone. After a few weeks of hard work, Pesho is now ready to present his creation.

The testing of the robot will be held on a rectangular table, composed of identical square cells. The table has **N** + 1 rows and **М** + 1columns, and every cell in the table is described by the pair (**x**,**y**) (0<= **x** <= **N,** 0 <= **y** <= **М**), denoting a cell on row **x** and column **y**.

It is known, that there are exactly **K** cells, from which the robot can takeoff or land on and that takeoff and landing take negligible amount of time.

For one unit of time the robot can go from point (**x,y**) to point (**x+dx,y+dy**) s.t. -1 <= **dx, dy** <= 1.

Pesho wants to present his work in the best possible way and he wants you to help him by finding the two closest cells for landing/takeoff and determine the time the robot will need to cover the distance between the two points.

**Input**

On the first row of the input file **robot.in** you are given three integers **N, M** and **К** - the number of rows, the number of columns and the number available cells for takeoff/landing.

On the each of the next **K** lines there are two numbers **Xi** and **Yi**, the coordinates (**Xi,Yi**) of the **i**-th cell.

**Output**

On the only row of the output file **robot.out** you should print one number - the minimal distance in units of time.

**Constraints:**

1 **N,М**  1,000,000,000

2 **K**  100,000

0 **Xi** **N**

0 **Yi** **M**

All the numbers in the input file are integers and there are no two points with the same coordinates.

**Time Limit: 2.5s**

**Example:**

|  |  |
| --- | --- |
| **robot.in** | **robot.out** |
| 5 5 3  3 1  1 3  4 4 | 2 |