

# Largest bunch of zeros

## Problem description

You are given a  $m \times n$  binary matrix (contains only 1s and 0s). You have to use dynamic programming to find the largest contiguous square containing only zeros in the matrix.

## Input format

The input will contain two lines. The first line will contain three integers  $m$   $n$   $x$  which denote the row size, column size, and number of 1s in the matrix. The second line will consist of  $x$  pairs of integers, each denoting the location of a 1 in the matrix.

## Output format

You are required to print out two pairs of coordinates, that denote the top left and the bottom right of the square formed by the contiguous set of zeros. If there are multiple solutions, print the rectangle sorted by the coordinates of the top left vertex, first along row number, then along column.

In case the solution is not present, print just one integer 0. In case the solution is just one square, print starting and ending as the same point (so you would print  $x$   $y$   $x$   $y$  if the square is located at  $x$   $y$  )

## Sample input

```
3 4 4
```

```
0 3 1 1 1 2 2 0
```

## Sample output

```
0 0 0 0
```

Notice that here there are two rectangles with the largest size, but the one printed is the one with top left vertex sorted according to row index, and then along column index.

## Scoring method

Please note that apart from the sample test case posted here, there will be other hidden test cases that your code will be checked on. The final score you see is a percentage of the test cases you have passed. If you pass only the sample test case, you will get 0/100 (even though your score will be shown as 10/100).

Each of these questions are finally worth the points mentioned in the assignment pdf file. So your credit for this problem shall be  $\text{your score}/100 * \text{points for this question}$ . Your final credit for programming assignment 3 shall be  $\text{final score} / 55 * 100$ .

## Plagiarism and copying

Strict action will be taken against students who are found to be indulging in plagiarism. Please note that changing variable names, removing indentation or moving code around will not help with regards to plagiarism checking.