

Chinese Checkers is a commercial variant of the older Scandinavian game called *Halma*. In this contest we propose a simplified version in a rectangular chessboard for two players.

Imagine a rectangular chessboard divided into L lines and C columns (usually C smaller than L). In each square (intersection of a line with a column) of that matrix there is a hole to support a piece of the game. The base of the rectangle (bottom) is line 1 and the top is line L ; columns are numbered from 1 to C from left to right.

Each player has as starting field (its “home”) at the bottom or at the top of the chess-board, and he has $P = 2 * C$ pieces (different colors for each players) that inserts into the holes of the two first rows of his field (lines 1 and 2 for the first Player, and lines $L - 1$ and L for the second Player). The main goal of each player is to invade the other player’s field with his own P pieces, moving just one piece each time. The winner is the player that first reaches the goal.

One piece can be moved from its position to a free hole on the left, on the right, or in front (moving only one step). Or else, it can jump over one other piece (belonging to any one of both players, colors doesn’t matter) in a straight line or along a diagonal to the next hole (right, left or in front) if it is free; this jump can continue as many times as it is possible in the same play (if in the new hole it is possible to jump over a piece to another free hole in the neighborhood).

The aim is to plan just one round for a selected piece of Player 1 (the one that starts at the base of the chess-board), given the position (coordinates of the hole) of the $2 * P$ pieces, at that precise moment of the game.

The question is to know all the movements that the selected piece can do; we just want to know the number of steps or jumps and the coordinates of the hole achieved. So your task is to write a program to read from an input file the status of the game and to produce the required plan.

Input

The input is composed of several test cases. Each one starts with a line containing two numbers defining, respectively, the number L of rows and the number C of columns in the chess-board. The following $2 * P$ lines, are the coordinates (line and column) of each piece. The last line of each case contains the coordinates of the selected piece, this is the piece of Player 1 for which we want to plan the possible movements.

Output

The output for each test case must be separated by a blank line.

Each test case must have one line for each possible movement, containing 3 numbers separated by one space. These numbers are: the coordinates of the destination hole (line and column) and the number of steps/jumps. The result must be sorted in decreasing order of lines and increasing order of columns. If there is more than one way for a single piece to reach a destination, output the minimum amount of steps/jumps needed to reach that destination.

Note: The following sample illustrates the status of the chessboard at the starting of the game: all pieces are at “home” (this is, in their initial holes in both starting fields). In that configuration, Player 1 wants to plan the movements for piece (1,4).

Sample Input

```
15 5
1 1
1 2
1 3
1 4
1 5
2 1
2 2
2 3
2 4
2 5
14 1
14 2
14 3
14 4
14 5
15 1
15 2
15 3
15 4
15 5
1 4
```

Sample Output

```
3 2 1
3 4 1
```