

The people of the remote country KnightLand are crazy about chess. They play chess whenever they get some free time. They have even divided their land into adjacent equal sized squares just like a chessboard. They like the knight of chess so much that they always go from one place to another following a knight's trail, even if it takes a lot of time and energy. In a minute, a KnightLander can make only one jump from one square to another like a knight.

For the past few years, the people of KnightLand are facing a serious problem. A monster has appeared in one of the squares of KnightLand. When it is awake, it eats anyone who enters its square. The good thing is, the monster doesn't move from one square to another and loves to sleep. If one enters its square while it is sleeping, it wakes up and forgets for a year how to sleep. It remains sleepy for a minute and cannot do any harm to anyone in this time. It then starts eating anyone it gets in its square. Therefore, the person who makes it awake can escape safely from it. But if one comes to that square later, he dies a terrible death.

In order to solve the problem, the 4 leaders of KnightLand are planning to meet together. They want to find a square that can be reached safely and as quickly as possible from the squares where they line in. To be extra careful, they decided that there should not be more than 1 jump in a minute in the whole KnightLand. Now, they want to find out if it is at all possible to meet at any square, and the minimum time required to get there.

Can you help them?

Input

Input consists of multiple test cases and terminated by an EOF. Each test case consists of 4 lines. The first line contains the string 'Set#*n*', where *n* is the set number. The second line contains number of rows *r* and number of columns *c* of squares in KnightLand's map (where $3 \leq r \leq 16$ and $3 \leq c \leq 16$). The third line contains position of each leader in the map, in terms of *row#* and *column#* of each. The fourth line gives the position of the monster. Note that the square at the upper left corner of the map has (*row#*, *column#*) = (1, 1). You may assume that no leader resides in the same square as the monster.

Output

For each set of input, there should be 2 lines of output. The first line should contain the string 'Set#*n*', where *n* is the set number. The second line should give the minimum time in minutes needed to reach a square (in format shown below), or the string 'Meeting is impossible.', whichever is applicable.

Sample Input

```
Set#1
5 5
1 1 1 5 5 1 4 4
3 3
Set#2
3 3
1 1 1 2 1 3 2 2
3 2
```

Sample Output

```
Set#1
Minimum time required is 6 minutes.
Set#2
Meeting is impossible.
```