Problem H. Sliding Blocks

Time limit 4000 ms
Mem limit 1048576 kB
OS Linux

Sliding Blocks is an interesting puzzle game which can be played on most mobile devices. In this game, you are given an $N \times M$ board with exactly one cell containing a block; all the other cells are empty. You can add a block to the board by sliding the new block from the edge of the board. The new block slides until it bumps into an existing block and it stops.

For example, let the board be 5×6 and there is a block at $(2, 3)$–the 2^{nd} row and 3^{rd} column.

```
......
..X...
......
......
......
```

If we slide a new block at the third column from the bottom edge, then the new block stops at $(3, 3)$ as it bumps into the block at $(2, 3)$.

```
......    ......
..X...    ..X...
......    ..X...
......    ......
......    ......
  ^
```

If we slide a new block at the second row from the right edge, at the fourth column from the top edge, and at the third row from the left edge, respectively, then the new blocks stop at $(2, 4)$, $(1, 4)$, and $(3, 2)$.

```
             v
......    ......    ...X..    ...X..
..X...<   ..XX..    ..XX..    ..XX..
..X...    ..X...   >..X...    .XX...
......    ......    ......    ......
......    ......    ......    ......
```

Note that a move which does not add a new block into the board is considered illegal, e.g. sliding a new block at the fourth column from the top edge or at the fifth row from the right edge in the last board of the previous example.

```
   v  [illegal]
...X..              ...X..
..XX..              ..XX..
```

```
.XX...             .XX...
......             ......
......             ......<  [illegal]
```

In each level of the game, you are given a target board which is your goal; in other words, you have to add zero or more new blocks such that your board becomes exactly the same as the target board. It is guaranteed that the target board contains blocks which form a tree (as in graph theory), i.e. from any block, there exists a unique simple path to any other block which only passes through other blocks. A block is connected to another block if they are next to each other.

The game seems good and you (the developer) are ready to enjoy the success. Until one day, an incoming user report states (and unfortunately, also proves) that a certain puzzle level is impossible to be solved. For example, consider the following initial board and target board.

```
....   XXXX
....   X..X
X...   X.XX
```

The right target board cannot be achieved from the initial board on the left as there is no way to add a new block at $(3, 3)$.

Now, you have to re-evaluate every puzzle in the game properly. Given the initial and target board configurations, determine whether it is possible to make your initial board become exactly the same as the target board.

Input

The first line contains three integers N, M, and B ($1 \leq N, M, B \leq 4 \cdot 10^5$) representing the board size (number of rows and columns) and the number of blocks in the target board, respectively. The next B lines each contains two integers r and c ($1 \leq r \leq N; 1 \leq c \leq M$) representing a block at the r^{th} row and c^{th} column in the target board. The first block which is given in the input (second row) already exists in your initial board. The target board configuration is guaranteed to contains only blocks which form a tree.

Output

The output contains a line denoting whether it is possible to make your board becomes exactly the same as the target board. If it is not possible, output "`impossible`" (without quotes) in one line. If it is possible, output "`possible`" (without quotes) in one line followed by $B - 1$ lines representing the moves which can achieve the goal. Each move is represented by a character c (c \in \{ \texttt{<},\texttt{>},\texttt{^},\texttt{v}\}) and an integer k, separated by a single space.

- < denotes sliding the new block from the right edge of the board towards the left.
- > denotes sliding the new block from the left edge of the board towards the right.
- \texttt{^} denotes sliding the new block from the bottom edge of the board upward.
- v denotes sliding the new block from the top edge of the board downward.

The integer k denotes the row or column in which the move is performed on. If $c \in \{<, >\}$, then the move is performed on the k^{th} row and $1 \le k \le N$. If c \in \{ \texttt{^},\texttt{v}\} , then the move is performed on the k^{th} column $1 \le k \le M$. Note that the moves are performed in the output order. If there is more than one move configuration which can achieve the goal, you may output any one of them.

Sample 1

Input	Output

3 4 6
1 1
1 2
2 2
2 3
3 3
3 4
	possible
	< 1
	^ 2
	< 2
	^ 3
	< 3

Sample 2

Input	Output

3 4 9
3 1
2 1
1 1
1 2
1 3
1 4
2 4
3 4
3 3
	impossible