Problem D - Jumping Hero

Background

A software house has decided to create a computer game, where the hero must find its way from a start position to the end position, through a labyrinth. In the labyrinth, some cells contain magic fountains that can be used to get super-powers an infinite number of times. Whenever the hero enters a cell with a magic fountain, he gets super-powers.

Usually, our hero moves in the labyrinth one cell left/right/up/down at a time (to an empty cell). With super-powers, the hero jumps to an empty cell N positions to the left/right/up/down. The super-power lasts for M jumps, and the hero can change its jumping direction after each jump. A jump is allowed if the end cell of the jump is within the map and it is not a wall – thus, the hero can jump over walls. If the hero jumps to a cell with a new magic fountain, the hero gets the super-powers of the new magic fountain, and the remaining effect of the previous magic fountain is cancelled. If the hero jumps to the cell where he obtained its current super-powers, no effect occurs (i.e., the hero gets no additional super-powers). When the current super-power ends, the hero proceeds its normal one-cell movement. If, after getting super-powers in some fountain, the hero cannot move to any cell, he looses his super-powers and returns to his previous cell. To reach the end position, the hero must move to the end cell or finish one jump in the end cell.

Problem

Given the labyrinth map compute the minimum number of moves/jumps from the start position to the end position.

Input

The first line of the input contains two positive integers, L and C, separated by a empty space, with L the number of lines and C the number of columns in the map. L and C are both lesser than 300. The following L lines of the input contain C integers each that define the cells of the map (separated by a empty space). Each integer, i, must be interpreted as follows: i = 0 represents a wall; i = 1 represents an empty cell (where the hero can move into); i = M*10+N represents an empty cell with a magic fountain that makes the hero jump M times to the cell that is N positions to the left/right/up/down of the current cell. M ranges from 1 to 5 and N ranges from 2 to 6. The maximum number of magic fountains in a map is 5,000. The two last lines of the input define the coordinates of the start position and end position (coordinates consist of two integers, denoting the line and column respectively, starting from 0).

Output

The output consists of one single line that contains an integer with the minimum number of moves/jumps, from the start position to the end position. If it is impossible to reach the end position, the output should be a single line containing `IMPOSSIBLE`.

Sample Input

```
8 8
0 1 1 1 1 1 1 1
0 1 0 0 1 13 1 1
0 1 32 1 1 1 0 0
0 1 1 0 1 1 1 0
0 1 1 0 0 0 0 0
0 1 1 1 1 1 1 0
0 1 0 0 1 1 1 0
0 1 1 1 1 1 1 0
1 7
5 4
```

Sample Output

14