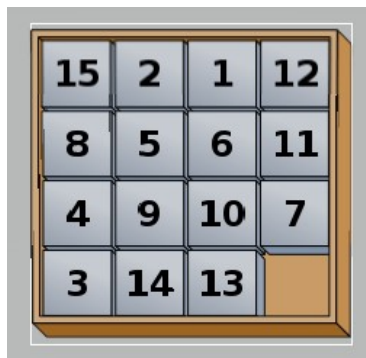


## E: Extended Puzzle

Source file name: `extended.c`, `extended.cpp`, `extended.java`, or `extended.py`

Author: Rodrigo Cardoso

The  $m \times n$ -puzzle, as explained in Wikipedia and Wolfram MathWorld, is a sliding puzzle with a frame of  $m$  rows and  $n$  columns, and a total of  $mn$  squared tiles. Each tile is assigned a unique integer number from 1 to  $mn$  and it is randomly placed in the frame. The tile assigned  $mn$  is removed from the frame, so that the neighboring tiles (horizontally and vertically) may slide into the empty square generating another puzzle configuration. Such a transformation is called a *move*. The figure below depicts a  $4 \times 4$ -puzzle where the tiles assigned 7 and 13 are the only possible moves:



A solution to an  $m \times n$ -puzzle is a sequence of moves that ends in a configuration with the tiles arranged in ascending order (i.e.,  $1, 2, \dots, mn$ ) when the rows are considered as a large array; the missing  $mn$  tile is assumed to be the last one. In this way, solving an  $m \times n$ -puzzle can be considered as a sorting process starting from the initial configuration, where each configuration can be represented as a permutation of  $1, 2, \dots, mn$ . Recall that a permutation may be classified either as *even* or *odd* depending on the *parity* of the number of inversions it contains: an *inversion* is a pair of elements in the permutation that are out of its usual order. In the picture above, for example, the tiles 8 and 10 are in the usual order, but 15 and 9 are out of the usual order.

A parity argument may be used to show that only half of the starting positions in the  $m \times n$ -puzzle are possible to solve (no matter how many moves are made). This is done by considering a function of the tile configuration invariant under any valid move, so that it partitions the space of all configurations into either *reachable* and *unreachable*. One such a function is the parity of the initial permutation plus the parity of the Manhattan distance (or “taxicab” distance) from the  $mn$  tile to its rightful position (i.e. from the empty square to the last column of the last row in the frame). Recall that the Manhattan distance is the minimum number of horizontal and vertical unit segments required to go from the starting square to the final one. In the above figure, the Manhattan distance from the 4 tile to the last column of the last row in the frame is 4, while it is 0 for the  $mn$  (i.e., the empty square) to the to the last column of the last row. This function is invariant because each move changes both the parity of the permutation and the parity of the taxicab distance. Note that, in particular, if the empty square is in the lower right corner then the puzzle is solvable if and only if the permutation of the remaining pieces is even. Although it is not so easy to see, the converse of the former claim is also true: a given configuration can be solved if and only if it represents a permutation whose parity plus that of the taxicab distance from the empty square to the lower right corner is even.

A very famous instance of the game is the  $4 \times 4$ -puzzle proposed by Sam Loyd in the 19th century in which each tile is placed correctly, except for the 14 and 15 ones that are swapped. Besides falsely claiming the game invention, he offered a \$1,000 prize for anyone who could provide a solution for that particular instance of

the game: of course, nobody won the prize because there is not such a solution since the initial configuration represents an odd permutation.

Your task is to establish if some given configurations of  $m \times n$ -puzzles are solvable or not.

### Input

The input consists of several test cases. Each test case begins with one line containing two blank-separated positive integers  $m$  and  $n$ ,  $1 < m \cdot n \leq 100\,000$ , representing the number of rows and columns of an  $m \times n$ -puzzle, respectively. Then  $m$  lines follow, each containing  $n$  numbers. The  $mn$  numbers listed in the  $m$  rows are a permutation of the integer numbers  $1, 2, \dots, mn$  and represent an initial configuration of an  $m \times n$ -puzzle.

*The input must be read from standard input.*

### Output

For each test case, output one line with 'Y' if the given configuration has a solution and 'N' otherwise.

*The output must be written to standard output.*

Sample Input	Sample Output
4 4	N
1 2 3 4	Y
5 6 7 8	Y
9 10 11 12	N
13 15 14 16	
2 2	
4 3	
2 1	
2 3	
4 1 3	
6 2 5	
2 3	
4 1 3	
6 5 2	