E. Table Compression

time limit per test: 4 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Little Petya is now fond of data compression algorithms. He has already studied *gz*, *bz*, *zip* algorithms and many others. Inspired by the new knowledge, Petya is now developing the new compression algorithm which he wants to name *dis*.

Petya decided to compress tables. He is given a table a consisting of n rows and m columns that is filled with positive integers. He wants to build the table a' consisting of positive integers such that the relative order of the elements in each row and each column remains the same. That is, if in some row i of the initial table $a_{i,j} < a_{i,k}$, then in the resulting table $a'_{i,j} < a'_{i,k}$, and if $a_{i,j} = a_{i,k}$ then $a'_{i,j} = a'_{i,k}$. Similarly, if in some column j of the initial table $a_{i,j} < a_{p,j}$ then in compressed table $a'_{i,j} < a'_{p,j}$ and if $a_{i,j} = a_{p,j}$ then $a'_{i,j} = a'_{p,j}$.

Because large values require more space to store them, the maximum value in a' should be as small as possible.

Petya is good in theory, however, he needs your help to implement the algorithm.

Input

The first line of the input contains two integers n and m (, the number of rows and the number of columns of the table respectively.

Each of the following n rows contain m integers $a_{i,j}$ $(1 \le a_{i,j} \le 10^9)$ that are the values in the table.

Output

Output the compressed table in form of n lines each containing m integers.

If there exist several answers such that the maximum number in the compressed table is minimum possible, you are allowed to output any of them.

Examples

input

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

output

```
1 2
2 3
```

input

```
4 3
20 10 30
50 40 30
50 60 70
90 80 70
```

output

```
2 1 3
5 4 3
5 6 7
9 8 7
```

Note

In the first sample test, despite the fact $a_{1,2} \neq a_{21}$, they are not located in the same row or column so they may become equal after the compression.