

Christmas Tree

Welcome aboard the spaceship The Circular! The crew is decorating its control panel: an $N \times M$ grid of lights, where each light has an integer brightness.

The grid is given as a matrix T . The value $T_{i,j}$ is the brightness in row i and column j . Rows are numbered from 1 to N from top to bottom, and columns from 1 to M from left to right.

A **Christmas tree** of height H ($H > 1$) is chosen by selecting a cell (i, j) as the star of the tree. The tree occupies exactly H rows: $i, i + 1, \dots, i + H - 1$. On row $i + d$ ($0 \leq d < H$), the tree contains all cells with columns $j - d, j - d + 1, \dots, j + d$.

The tree must fit entirely inside the grid. Also, for every two consecutive rows of the tree, every brightness value in the upper row of the tree must be strictly smaller than every brightness value in the lower row of the tree.

More formally,

the star of a tree of height H must be located in a cell (i, j) such that $1 \leq i \leq N - H + 1$ and $H \leq j \leq M - H + 1$.

for every $d = 0, 1, \dots, H - 1$, let

$$S_d = \{T_{i+d, j-d}, T_{i+d, j-d+1}, \dots, T_{i+d, j+d}\}.$$

Then for every $r = 0, 1, \dots, H - 2$, every $x \in S_r$ and every $y \in S_{r+1}$ must satisfy $x < y$.

For every height k from 2 to N , determine the maximum possible value of the star $T_{i,j}$ such that there exists a valid Christmas tree of height exactly k whose star is at (i, j) , or output -1 if it doesn't exist.

Input

The first line contains two integers N and M .

The next N lines each contain M integers describing the matrix T .

Note: the input might be large, so it is recommended to use

```
ios::sync_with_stdio(false);  
cin.tie(0);
```

Output

Print $N - 1$ lines.

On the line corresponding to $k = 2, 3, \dots, N$, output the answer for height k : the maximum star value among all valid Christmas trees of height exactly k , or -1 if no such tree exists.

Constraints

- $2 \leq N \leq 2500$
- $1 \leq M \leq 2500$
- $1 \leq T_{i,j} \leq 10^9$

Subtasks

Subtask 1 (3 points)

$$M < 3$$

Subtask 2 (5 points)

$$N = 3$$

Subtask 3 (12 points)

$$N, M \leq 50$$

Subtask 4 (15 points)

$$N, M \leq 200$$

Subtask 5 (19 points)

$$N, M \leq 600$$

Subtask 6 (38 points)

$$N, M \leq 900$$

Subtask 7 (8 points)

No additional constraints

Examples

Input 1

```
4 5
5 1 4 2 7
8 9 10 11 12
18 14 20 16 17
1 1 1 1 1
```

Output 1

```
11
4
-1
```

Explanation 1

For height 2, the best possible star is the cell with brightness 11 in the second row and fourth column. The row below it contains brightness values 20, 16, and 17, which are all larger. 12 can't be the star, since the tree would not fit inside the grid.

5	1	4	2	7
8	9	10	11	12
18	14	20	16	17
1	1	1	1	1

For height 3, the best possible star has brightness 4.

5	1	4	2	7
8	9	10	11	12
18	14	20	16	17
1	1	1	1	1

No valid tree of height 4 exists.