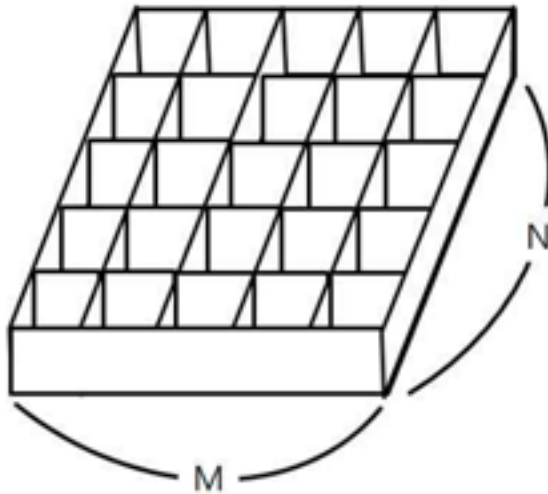


Offline Coding Test

John's tomato farm has a large warehouse that keeps tomatoes, which, as shown below, is stored in a warehouse by putting one in a lattice box.



Some tomatoes stored in the warehouse are ripe, but some tomatoes may not yet be ripe. After a day to put in the box ($M * N$) from the storage, unripe tomatoes adjacent to the ripe tomatoes are ripe under the influence of ripe tomatoes. The adjacent place of one tomato means the tomato in the four directions: left, right, front, and back. Suppose that there is no effect on the diagonal tomatoes and that tomatoes do not ripen on their own.

John wants to know how many days it will take for all tomatoes to be ripen.

The size of the lattice-shaped boxes that store tomatoes in the warehouse and the information of ripe tomatoes and unripe tomatoes are given; some may not contain tomatoes.

Create a program to find out how many days it takes for all tomatoes to ripen.

Input

Two integers M , N representing the size of the box are given to the first line; M represents the number of horizontal spaces of the box, N represents the number of vertical spaces of the box; $2 \leq M$, $N \leq 1,000$.

From the second line, the information of the tomatoes stored in one box is given, that is, from the second line to the N line, the information of the tomatoes in the box is given. In one line, the state of tomatoes in the box horizontal line is given as M integers.

Number 1 represents ripe tomatoes, number 0 represents unripe tomatoes, and number -1 represents empty spaces.

Output

You have to print out the minimum date until the tomatoes are all ripe. If all tomatoes are already ripe, you have to output zero, and if the tomatoes are not ripe no matter how long, you have to output -1.

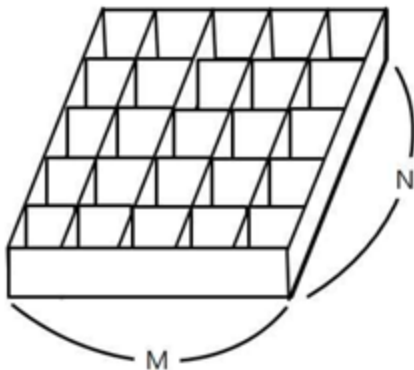
Sample test sets

Input	Output
6 4 0 1	8
6 4 0 -1 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1	-1
6 4 1 -1 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 -1 1	6
5 5 -1 1 0 0 0 0 -1 -1 -1 0 0 -1 -1 -1 0 0 -1 -1 -1 0 0 0 0 0 0	14
2 2 1 -1 -1 1	0

Tomato

Korean

철수의 토마토 농장에서는 토마토를 보관하는 큰 창고를 가지고 있다. 토마토는 아래의 그림과 같이 격자 모양 상자의 칸에 하나씩 넣어서 창고에 보관한다.



창고에 보관되는 토마토들 중에는 잘 익은 것도 있지만, 아직 익지 않은 토마토들도 있을 수 있다. 보관 후 하루가 지나면, 익은 토마토들의 인접한 곳에 있는 익지 않은 토마토들은 익은 토마토의 영향을 받아 익게 된다. 하나의 토마토의 인접한 곳은 왼쪽, 오른쪽, 앞, 뒤 네 방향에 있는 토마토를 의미한다. 대각선 방향에 있는 토마토들에게는 영향을 주지 못하며, 토마토가 혼자 저절로 익는 경우는 없다고 가정한다. 철수는 창고에 보관된 토마토들이 며칠이 지나면 다 익게 되는지, 그 최소 일수를 알고 싶어 한다.

토마토를 창고에 보관하는 격자모양의 상자들의 크기와 익은 토마토들과 익지 않은 토마토들의 정보가 주어졌을 때, 며칠이 지나면 토마토들이 모두 익는지, 그 최소 일수를 구하는 프로그램을 작성하라. 단, 상자의 일부 칸에는 토마토가 들어있지 않을 수도 있다.

입력

첫 줄에는 상자의 크기를 나타내는 두 정수 M, N 이 주어진다. M 은 상자의 가로 칸의 수, N 은 상자의 세로 칸의 수를 나타낸다. 단, $2 \leq M, N \leq 1,000$ 이다. 둘째 줄부터는 하나의 상자에 저장된 토마토들의 정보가 주어진다. 즉, 둘째 줄부터 N 개의 줄에는 상자에 담긴 토마토의 정보가 주어진다. 하나의 줄에는 상자 가로줄에 들어있는 토마토의 상태가 M 개의 정수로 주어진다. 정수 1은 익은 토마토, 정수 0은 익지 않은 토마토, 정수 -1은 토마토가 들어있지 않은 칸을 나타낸다.

출력

여러분은 토마토가 모두 익을 때까지의 최소 날짜를 출력해야 한다. 만약, 저장될 때부터 모든 토마토가 익어있는 상태이면 0을 출력해야 하고, 토마토가 모두 익지는 못하는 상황이면 -1을 출력해야 한다.

Sample test sets

Input	Output
<pre> 6 4 0 1 </pre>	8
<pre> 6 4 0 -1 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 </pre>	-1
<pre> 6 4 1 -1 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 -1 1 </pre>	6
<pre> 5 5 -1 1 0 0 0 0 -1 -1 -1 0 0 -1 -1 -1 0 </pre>	14

0 -1 -1 -1 0 0 0 0 0 0	
2 2 1 -1 -1 1	0