

Task: POW

The Flood



Stage II. Day 1. Source file `pow.*`

07.02.2007

Available memory: 128 MB.

Byteburg, the capital of Byteotia, is a picturesque city situated in a valley in the midst of mountains. Unfortunately, recent heavy rainfall has caused a flood — all the Byteburg is now completely under water. Byteasar, the king of Byteotia, has summoned his most enlightened advisors, including you, to a council. After long deliberations the council agreed to bring a few pumps, set them up in the flooded area and drain Byteburg. The king has asked you to determine the minimum number of pumps sufficing to drain the city.

You are provided with a map of the city and the valley it is situated in. The map is in the shape of a $m \times n$ rectangle, divided into unitary squares. For each such square the map tells its height above sea level and also whether it is a part of Byteburg or not. The whole area depicted in the map is under water. Furthermore, it is surrounded by much higher mountains, making the outflow of water impossible. Obviously, there is no need to drain the area that does not belong to Byteburg.

Each pump can be placed in any unitary square depicted in the map. The pump will be drawing the water until its square is completely drained. Of course, the communicating tubes principle makes its work, so draining one square results in lowering the water level or complete draining of those squares from which the water can flow down to the one with the pump. Water can flow only between squares with a common side (or, more exact, squares whose projections onto horizontal plane have a common side, since the squares may be at different level). Apart from that, the water obviously only flows down.

Task

Write a programme that:

- reads description of the map from the standard input,
- determines the minimum number of pumps needed to drain whole Byteburg,
- writes out the outcome to the standard output.

Input

In the first line of the standard input there are two integers m and n , separated by a single space, $1 \leq m, n \leq 1000$. The following m lines contain the description of the map. The $(i + 1)^{th}$ line describes the i^{th} row of unitary squares in the map. It contains n integers $x_{i,1}, x_{i,2}, \dots, x_{i,n}$, separated by single spaces, $-1000 \leq x_{i,j} \leq 1000, x_{i,j} \neq 0$. The number $x_{i,j}$ describes the j^{th} square of the i^{th} line. The ground level in this square is $|x_{i,j}|$ above sea level. If $x_{i,j} > 0$, then the square is part of Byteburg, otherwise it is outside the city. Notice, that the area of Byteburg need not be connected. In fact the city may have several separate parts.

Output

Your programme should write out one integer to the standard output — the minimum number of pumps needed to drain Byteburg.

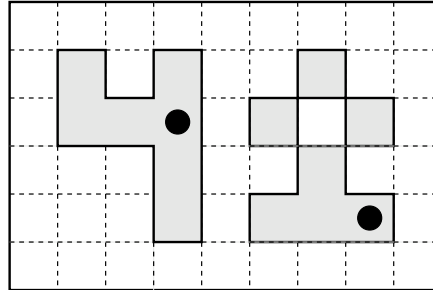
Example

For the input data:

```
6 9
-2 -2 -1 -1 -2 -2 -2 -12 -3
-2 1 -1 2 -8 -12 2 -12 -12
-5 3 1 1 -12 4 -6 2 -2
-5 -2 -2 2 -12 -3 4 -3 -1
-5 -6 -2 2 -12 5 6 2 -1
-4 -8 -8 -10 -12 -8 -6 -6 -4
```

the correct result is:

2



In the figure you can see the area of Byteburg and an exemplary setup of two pumps.