

Ladrilhos

Avelino tem um mosaico em uma das paredes de sua casa. É um mosaico muito antigo, composto por pequenos ladrilhos coloridos. Como é um mosaico antigo, alguns ladrilhos se soltaram ao longo dos anos formando buracos.

Agora, Avelino quer restaurar o mosaico cobrindo os buracos com novos ladrilhos. Entretanto, para economizar, Avelino quer comprar ladrilhos de uma única cor para tapar os buracos. Em particular, quer comprar ladrilhos de uma das cores originais ou de uma cor ainda não contida no mosaico.

Por ser um mosaico, não se deseja que hajam áreas muito grandes com a mesma cor. Avelino resolveu que vai escolher a cor dos ladrilhos tentando fazer com que o tamanho da menor área monocromática seja o menor possível, para que haja mais detalhes. Veja que pode existir mais de uma cor possível. Uma área é monocromática se todos os ladrilhos nela são da mesma cor. Dois ladrilhos adjacentes fazem parte da mesma área se possuem a mesma cor, e dois ladrilhos são adjacentes se compartilham um lado.

Veja o primeiro caso de exemplo, temos três áreas da cor 1 (uma de tamanho 3 e duas de tamanho 2), uma área da cor 2 (de tamanho 3) e uma área da cor 3 de tamanho 7. Uma resposta possível seria escolher a cor 2, fazendo com que a menor área monocromática seja de tamanho 2. Se escolhermos a cor 1 a menor área seria de tamanho 3.

Crie um programa que imprima o tamanho da menor área possível.

Entrada

A primeira linha contém dois inteiros **H** e **L**, a altura e largura do mosaico, respectivamente, satisfazendo $1 \leq H \leq 200$ e $1 \leq L \leq 200$. Em seguida, **H** linhas conterão cada uma **L** inteiros, separados por espaço, correspondendo às cores dos ladrilhos. Um inteiro 0 corresponde a um buraco e um inteiro $i \neq 0$ corresponde a um ladrilho com a *i*-ésima cor, podendo ir de 1 até 40000 no máximo.

Saída

Seu programa deve produzir uma linha, contendo um inteiro, o tamanho da menor área possível.

Exemplo

Entrada:

```
3 8
3 3 3 1 1 0 0 0
3 1 1 0 2 2 0 1
3 3 3 0 0 2 1 1
```

Saída:

```
2
```

Entrada:

```
3 7
1 1 0 2 2 1 1
1 1 0 2 2 1 1
1 1 0 0 3 3 3
```

Saída:

```
3
```

Entrada:

```
3 6
2 2 2 2 0 2
```

2 2 2 0 2 2
2 2 2 2 0 2

Saída:
1