

## Question 1

Correct

Marked out of 10.00

You are given an  $m \times n$  matrix such that  $n = m + 1$ . In the given matrix, find if any number is consecutive for 3 times either in row, column, diagonals print the num.

If there are multiple such numbers, then print minimum of those numbers.

Input format

- First line contains m, the number of rows
- Following m lines contain n numbers

Example Input

```
6
2 3 4 5 6 2 4
2 3 4 7 6 7 6
2 3 5 5 5 5 2
2 3 1 1 2 1 3
1 1 1 1 9 0 3
2 3 1 1 5 1 2
```

Output

```
1
```

For example:

Input	Result
6 2 3 4 5 6 2 4 2 3 4 7 6 7 6 2 3 5 5 5 5 2 2 3 1 1 2 1 3 1 1 1 1 9 0 3 2 3 1 1 5 1 2	1

Answer: (penalty regime: 0 %)

```
1 def consec_num(matrix,m,n):
2     min_num=float('inf')
3
4     for i in range(m):
5         for j in range(n-2):
6             if matrix[i][j] == matrix[i][j + 1] == matrix[i][j+2]:
7                 min_num = min(min_num,matrix[i][j])
8
9     for j in range(n):
10        for i in range(m-2):
11            if matrix[i][j] == matrix[i+1][j]==matrix[i+2][j]:
12                min_num=min(min_num,matrix[i][j])
13
14        for i in range(m-2):
15            for j in range(n-2):
16                if matrix[i][j]==matrix[i+1][j+1]==matrix[i+2][j+2]:
17                    min_num=min(min_num,matrix[i][j])
18
19        for i in range(m-2):
20            for j in range(2,n):
21                if matrix[i][j]==matrix[i+1][j-1]==matrix[i+2][j-2]:
22                    min_num=min(min_num,matrix[i][j])
23
24        return min_num if min_num!= float('inf') else -1
25
26
27 m=int(input())
28 matrix=[list(map(int,input().split())) for _ in range(m)]
29 n=m+1
30
```

```
30
31 | print(consec_num(matrix,m,n))
```

	Input	Expected	Got	
✓	6 2 3 4 5 6 2 4 2 3 4 7 6 7 6 2 3 5 5 5 5 2 2 3 1 1 2 1 3 1 1 1 1 9 0 3 2 3 1 1 5 1 2	1	1	✓

Passed all tests! ✓