T 113 - Johny in a Matrix

Problem Submissions

You are given a matrix A of size NxN. Every cell has a value A[i][j] associated with it. Johny is standing currently at position (1,1) and he intends to get to cell (N,N).

The cost of going from cell (i,j) to cell(x,y) is A[i][j] * A[x][y].

Johny can either decided to travel in a row-major order or in a column-major order.

Your task is to find out which method will cost him less. Print "row-major" or "column-major" accordingly.

Input

First line contains an integer N
Next N lines each contain N integers each denoting the value of the array A[][]

Output

Solved: 716
Attempted: 722

Print either "row-major" or "column-major". If both ways cost the same then print "row-major".

Sample Input 0

```
2
1 2
3 4
```

Sample Output 0

```
column-major
```

Explanation 0

row major cost = 1x2 + 2x3 + 3x4 = 2 + 6 + 12 = 20
col major cost = 1x3 + 3x2 + 2x4 = 3 + 6 + 8 = 17
Hence the answer is column-major

Current Buffer (saved locally, editable) Python 3

```
1 size=int(input())
2 matrix=list()
3 row=0
4 col=0
5 mn=0
6 for i in range(size):
7     v=list(map(int,input().split()))
8     matrix.append(v)
```

```
for i in range(size):
    for j in range(size-1):
        row=row+matrix[i][j]*matrix[i][j+1]
for i in range(size-1):
    row=row+matrix[i][-1]*matrix[i+1][0]
for i in range(size):
    for j in range(size-1):
        col=col+matrix[j][i]*matrix[j+1][i]
for i in range(size-1):
    col=col+matrix[size-1][i]*matrix[0][i+1]
if(col<row):
    print('column-major')
else:
    print('row-major')
```

Line: 22 Col: 23

Upload Code as File Test against custom input Run Code Submit Code