

PROBLEMS SUBMIT STATUS STANDINGS CUSTOM TEST

A. Beautiful Matrix

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You've got a 5×5 matrix, consisting of 24 zeroes and a single number one. Let's index the matrix rows by numbers from 1 to 5 from top to bottom, let's index the matrix columns by numbers from 1 to 5 from left to right. In one move, you are allowed to apply one of the two following transformations to the matrix:

- Swap two neighboring matrix rows, that is, rows with indexes i and $i + 1$ for some integer i ($1 \leq i < 5$).
- Swap two neighboring matrix columns, that is, columns with indexes j and $j + 1$ for some integer j ($1 \leq j < 5$).

You think that a matrix looks *beautiful*, if the single number one of the matrix is located in its middle (in the cell that is on the intersection of the third row and the third column). Count the minimum number of moves needed to make the matrix beautiful.

Input

The input consists of five lines, each line contains five integers: the j -th integer in the i -th line of the input represents the element of the matrix that is located on the intersection of the i -th row and the j -th column. It is guaranteed that the matrix consists of 24 zeroes and a single number one.

Output

Print a single integer — the minimum number of moves needed to make the matrix beautiful.

Examples

input

Copy

0 0 0 0 0
0 0 0 0 1
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

output

Copy

3

input

Copy

0 0 0 0 0
0 0 0 0 0
0 1 0 0 0
0 0 0 0 0
0 0 0 0 0

output

Copy

1

→ Attention

The package for this problem was not updated by the problem writer or Codeforces administration after we've upgraded the judging servers. To adjust the time limit constraint, a solution execution time will be multiplied by 2. For example, if your solution works for 400 ms on judging servers, then the value 800 ms will be displayed and used to determine the verdict.

Codeforces Round #161 (Div. 2)

Finished

Practice



→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language:

Python 3.8.10

Almost always, if you send a solution on PyPy, it works much faster

Choose file:

Choose File

 No file chosen

Submit

→ Problem tags

implementation *800

No tag edit access

→ Contest materials

- Announcement
- Tutorial

[Codeforces](#) (c) Copyright 2010-2022 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Aug/06/2022 20:07:22^{UTC+4.5} (g2).
Desktop version, switch to [mobile version](#).
[Privacy Policy](#)

Supported by



ITMO UNIVERSITY