Rolling Stones

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

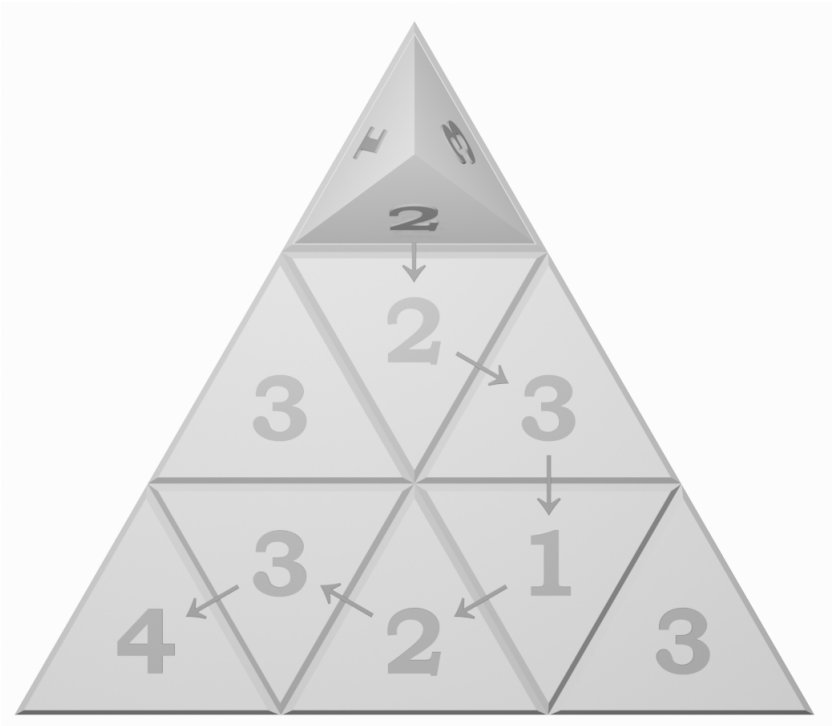
Bobo has been playing a puzzle called *Rolling Stones*, which takes place on an equilateral triangular board consisting of n ($n \geq 2$) rows and n^2 cells. Each cell on the board is labeled with a number from 1 to 4. Bobo also has a tetrahedral stone, with each face numbered from 1 to 4 (a tetrahedral dice), initially placed at the first cell in the first row of the board. The position of the stone is as follows: the face with the number 1 is towards the left, the face with the number 2 is towards the next row, the face with the number 3 is towards the right, and the face with the number 4 is on the bottom side.

The goal of the puzzle is to roll the stone to a target cell under the following rules:

- **Matching Numbers**: When the stone rests on a cell, the number on the cell must match the number on the stone's bottom face.
- **Single Visit**: Each cell can only be visited once throughout the journey, including the starting and target cells.

The stone rolls by tipping along an edge that touches the board, moving it to a neighboring cell. Given the board layout, the target cell, and the stone's initial orientation, Bobo wants to know: is it possible to reach the target cell following the rules? If possible, what is the minimum number of rolls required to reach the target?

The illustration for a solution of the first sample test is given as follows.



Illustration

Input

The first line contains an integer n ($2 \leq n \leq 100$), denoting the size of the board.

Then n lines follows, with the i-th ($1 \leq i \leq n$) line containing $2i-1$ numbers $a_{i,1}, a_{i,2}, \ldots, a_{i,2i-1}$, where each $1 \leq a_{i,j} \leq 4$ indicates the number on the j-th cell from left to right in the i-th row. It is guaranteed that $a_{1,1} = 4$.

Then another line follows, containing two integers x, y ($2 \leq x \leq n, 1 \leq y \leq 2x-1$). Here, (x, y) represents the target cell, located at the y-th cell from left to right in the x-th row.

Output

If there is no way to roll the stone to the target cell, output -1 in a line. Otherwise, output the minimum number of rolls to roll the stone to the target cell in a line.

Examples

standard input	standard output
3 4 3 2 3 4 3 2 1 3 3 1	6
3 4 3 3 3 4 3 2 1 3 3 1	-1