

D. Dasha and Chess

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

This is an interactive task.

Dasha and NN like playing chess. While playing a match they decided that normal chess isn't interesting enough for them, so they invented a game described below.

There are 666 black rooks and 1 white king on the chess board of size 999×999 . The white king wins if he gets checked by rook, or, in other words, if he moves onto the square which shares either a row or column with a black rook.

The sides take turns, starting with white. NN plays as a white king and on each of his turns he moves a king to one of the squares that are adjacent to his current position either by side or diagonally, or, formally, if the king was on the square (x, y) , it can move to the square (nx, ny) if and only $\max(|nx - x|, |ny - y|) = 1$, $1 \leq nx, ny \leq 999$. NN is also forbidden from moving onto the squares occupied with black rooks, however, he can move onto the same row or column as a black rook.

Dasha, however, neglects playing by the chess rules, and instead of moving rooks normally she moves one of her rooks on any space devoid of other chess pieces. It is also possible that the rook would move onto the same square it was before and the position wouldn't change. However, she can't move the rook on the same row or column with the king.

Each player makes 2000 turns, if the white king wasn't checked by a black rook during those turns, black wins.

NN doesn't like losing, but thinks the task is too difficult for him, so he asks you to write a program that will always win playing for the white king. Note that Dasha can see your king and play depending on its position.

Input

In the beginning your program will receive 667 lines from input. Each line contains two integers x and y ($1 \leq x, y \leq 999$) — the piece's coordinates. The first line contains the coordinates of the king and the next 666 contain the coordinates of the rooks. The first coordinate denotes the number of the row where the piece is located, the second denotes the column. It is guaranteed that initially the king isn't in check and that all pieces occupy different squares.

Output

After getting king checked, you program should terminate immediately without printing anything extra.

Interaction

To make a move with the king, output two integers x and y ($1 \leq x, y \leq 999$) — the square to which the king would be moved. The king cannot move onto the square already occupied by a rook. It is guaranteed that the king would always have a valid move.

After each of your turns read the rook's turn in the following format: a single line containing three integers k, x and y ($1 \leq k \leq 666$, $1 \leq x_i, y_i \leq 999$) — the number of the rook that would move and the square it would move to. It is guaranteed that the rook wouldn't move to a square already occupied by another chess piece, but it can move onto the square where it was before the turn so that its position wouldn't change. It is guaranteed that the move does not put your king into a check. If your king got in check, all three integers would be equal to -1 and in that case your program should terminate immediately.

After printing your turn do not forget to output end of line and flush the output. Otherwise you will get `Idleness limit exceeded`. To do this, use:

- `fflush(stdout)` or `cout.flush()` in C++;

Codeforces Round #532 (Div. 2)

Finished

→ 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 ACM-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

→ Problem tags

constructive algorithms games
 interactive *2400

No tag edit access

→ Contest materials

- Announcement #1 (en) ✕
- Announcement #2 (ru) ✕
- Tutorial #1 (en) ✕
- Tutorial #2 (ru) ✕
- Tutorial #3 (en) ✕

- `System.out.flush()` in Java;
- `flush(output)` in Pascal;
- `stdout.flush()` in Python;
- see documentation for other languages.

Answer "0 0 0" instead of a correct answer means that you made an invalid query. Exit immediately after receiving "0 0 0" and you will see `Wrong answer` verdict. Otherwise you can get an arbitrary verdict because your solution will continue to read from a closed stream.

Hacks are not allowed for this problem.

Example

input	Copy
999 999 1 1 1 2 2 1 2 2 1 3 2 3 <...> 26 13 26 14 26 15 26 16 1 700 800 2 1 2 <...> -1 -1 -1	
output	Copy
999 998 999 997 <...> 999 26	

Note

The example is trimmed. The full initial positions of the rooks in the first test are available at <https://pastebin.com/qQCTXgKP>. It is not guaranteed that they will behave as in the example.

[Codeforces](#) (c) Copyright 2010-2019 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Jul/22/2019 15:29:20^{UTC+8} (e1).
Desktop version, switch to [mobile version](#).
[Privacy Policy](#).

Supported by



ITMO UNIVERSITY