AutumnKite | Logout

HOME TOP CONTESTS GYM PROBLEMSET GROUPS RATING API HELP HONORCUP DASHA CALENDAR

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS ROOM STANDINGS CUSTOM INVOCATION

E2. Knightmare (hard)

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

This is an interactive problem.

Alice and Bob are playing a game on the chessboard of size $n \times m$ where n and m are **even**. The rows are numbered from 1 to n and the columns are numbered from 1 to m. There are two knights on the chessboard. A white one initially is on the position (x_1, y_1), while the black one is on the position (x_2, y_2). Alice will choose one of the knights to play with, and Bob will use the other one.

The Alice and Bob will play in turns and whoever controls **the white** knight starts the game. During a turn, the player must move their knight adhering the chess rules. That is, if the knight is currently on the position (x, y), it can be moved to any of those positions (as long as they are inside the chessboard):

$$(x+1, y+2), (x+1, y-2), (x-1, y+2), (x-1, y-2),$$
$$(x+2, y+1), (x+2, y-1), (x-2, y+1), (x-2, y-1).$$

We all know that knights are strongest in the middle of the board. Both knight have a single position they want to reach:

- the owner of the white knight wins if it captures the black knight or if the white knight is at $(n/2, m/2)$ and this position is not under attack of the black knight at this moment;
- The owner of the black knight wins if it captures the white knight or if the black knight is at $(n/2+1, m/2)$ and this position is not under attack of the white knight at this moment.

Formally, the player who captures the other knight wins. The player who is at its target square ($(n/2, m/2)$ for white, $(n/2+1, m/2)$ for black) and this position is not under opponent's attack, also wins.

A position is under attack of a knight if it can move into this position. Capturing a knight means that a player moves their knight to the cell where the opponent's knight is.

If Alice made 350 moves and nobody won, the game is a draw.

Alice is unsure in her chess skills, so she asks you for a help. Choose a knight and win the game for her. It can be shown, that Alice always has a winning strategy.

Interaction

The interaction starts with two integers n and m ($6 \le n, m \le 1000$, n and m are even) — the dimensions of the chessboard.

The second line contains four integers x_1, y_1, x_2, y_2 ($1 \le x_1, x_2 \le n$, $1 \le y_1, y_2 \le m$) — the positions of the white and the black knight. It is guaranteed that the two knights have different starting positions. It is also guaranteed that none of the knights are in their own target square in the beginning of the game (however, they can be on the opponent's target position).

Your program should reply with either "`WHITE`" or "`BLACK`", depending on the knight you want to play with. In case you select the white knight, you start the game.

During every your turn, you need to print two integers: x and y, the position to move the knight. If you won the game by this turn, you must terminate your program immediately.

After every turn of the opponent, you will receive two integers: x and y, the position where Bob moved his knight.

If your last move was illegal or you lost the game after jury's turn, or you made 350 moves, and haven't won, you will receive "`-1 -1`". In such cases, you should terminate your program

Codeforces Round #577 (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

→ Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language: GNU G++11 5.1.0 ▼

Choose file: 选择文件 未选择任何文件

Be careful: there is 50 points penalty for submission which fails the pretests or resubmission (except failure on the first test, denial of judgement or similar verdicts). "Passed pretests" submission verdict doesn't guarantee that the solution is absolutely correct and it will pass system tests.

Submit

→ Problem tags

graphs interactive shortest paths *3000

No tag edit access

→ Contest materials

- Announcement (en)

and then you will get a `Wrong Answer` verdict.

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

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

Hacks are disabled for this problem.

Jury's program is *adaptive*: the moves of jury may depend on the moves made by your program.

Examples

input Copy

```
8 8
2 3 1 8
```

output Copy

```
WHITE
4 4
```

input Copy

```
6 6
4 4 2 2
6 3
```

output Copy

```
BLACK
4 3
```

Note

In the first example, the white knight can reach it's target square in one move.

In the second example black knight wins, no matter what white knight moves.

- Tutorial #1 (en) ×
- Tutorial #2 (en) ×