UCC Coding Competition '21 P5 - Woodcutting Game

Time limit: 2.0s   **Memory limit:** 256M

After tearing down your pig barn, you and your friend Tyler have to dispose of two long rectangular wooden boards. You have to cut the boards into smaller pieces. Because this is boring, you devise a game to play to make things more interesting.

The two 2-dimensional rectangular boards have **heights of either** 1 **or** 2 and integer widths. For example, one board can be a 2×2021 board, and another can be a 1×5 board.

You and Tyler will take turns cutting the rectangular boards, and you will go first. Each of the cuts you make must turn one rectangular board into two smaller rectangular boards, each with an integer height and width. The cuts must be made either horizontally or vertically.

In particular, when it is your turn, you have two options as follows. Note that both options leave your opponent with two rectangles, so the game can continue.

Option 1: Choose one board to discard entirely, and then cut the other board into two smaller integer-dimension rectangles however you want.

The below diagram illustrates this option (but does not include all possible ways to cut the rectangle).
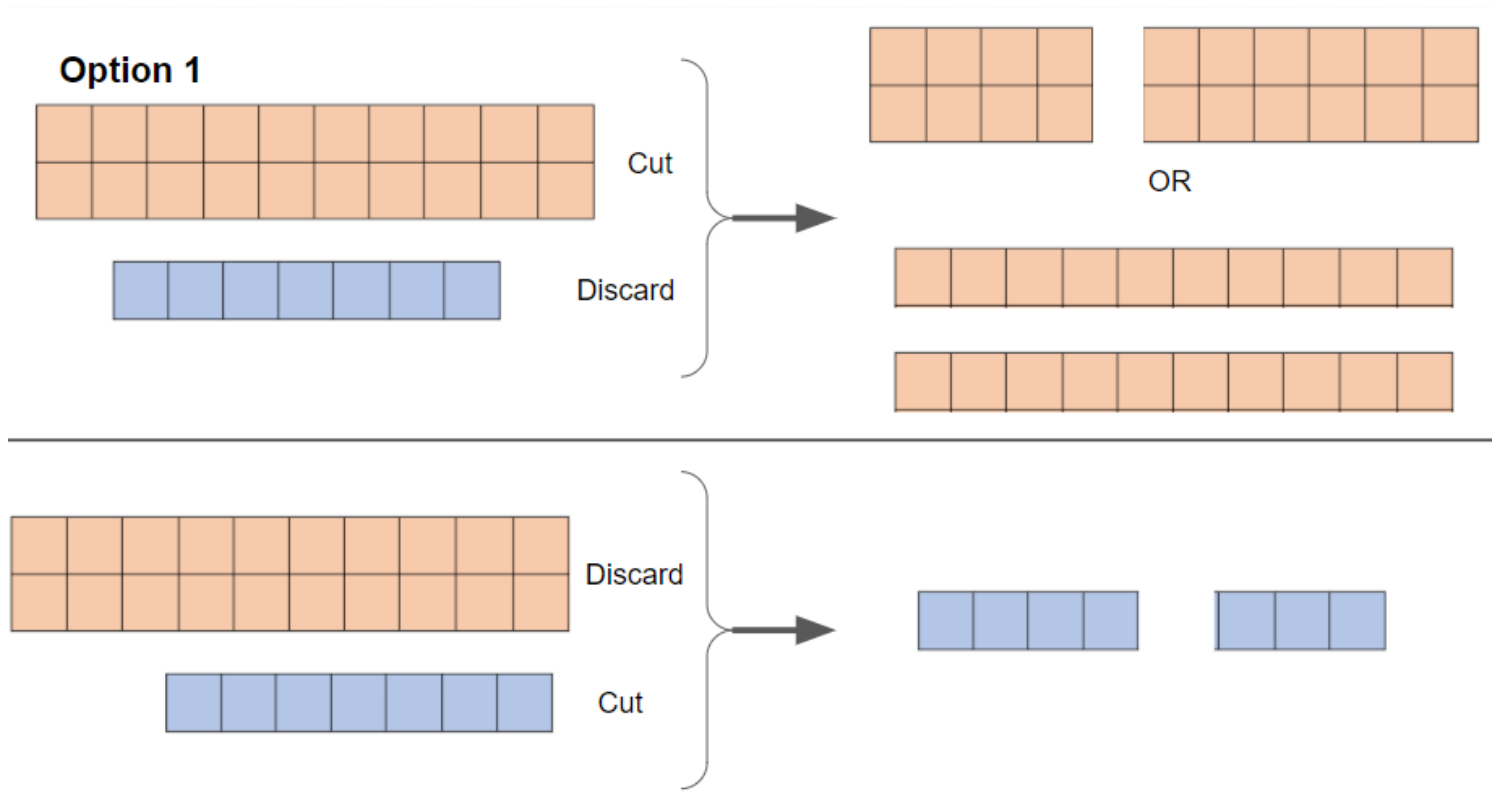
Option 2: Choose one board to leave untouched, cut a rectangular piece off the other board, and discard that piece.

If you choose Option 2, you must **EITHER cut 1 off the height OR cut any integer between 1 and 10 (inclusive) off the width** of the board that you have chosen to cut.

The below diagram illustrates this option (but does not include all possible ways to cut the rectangle).
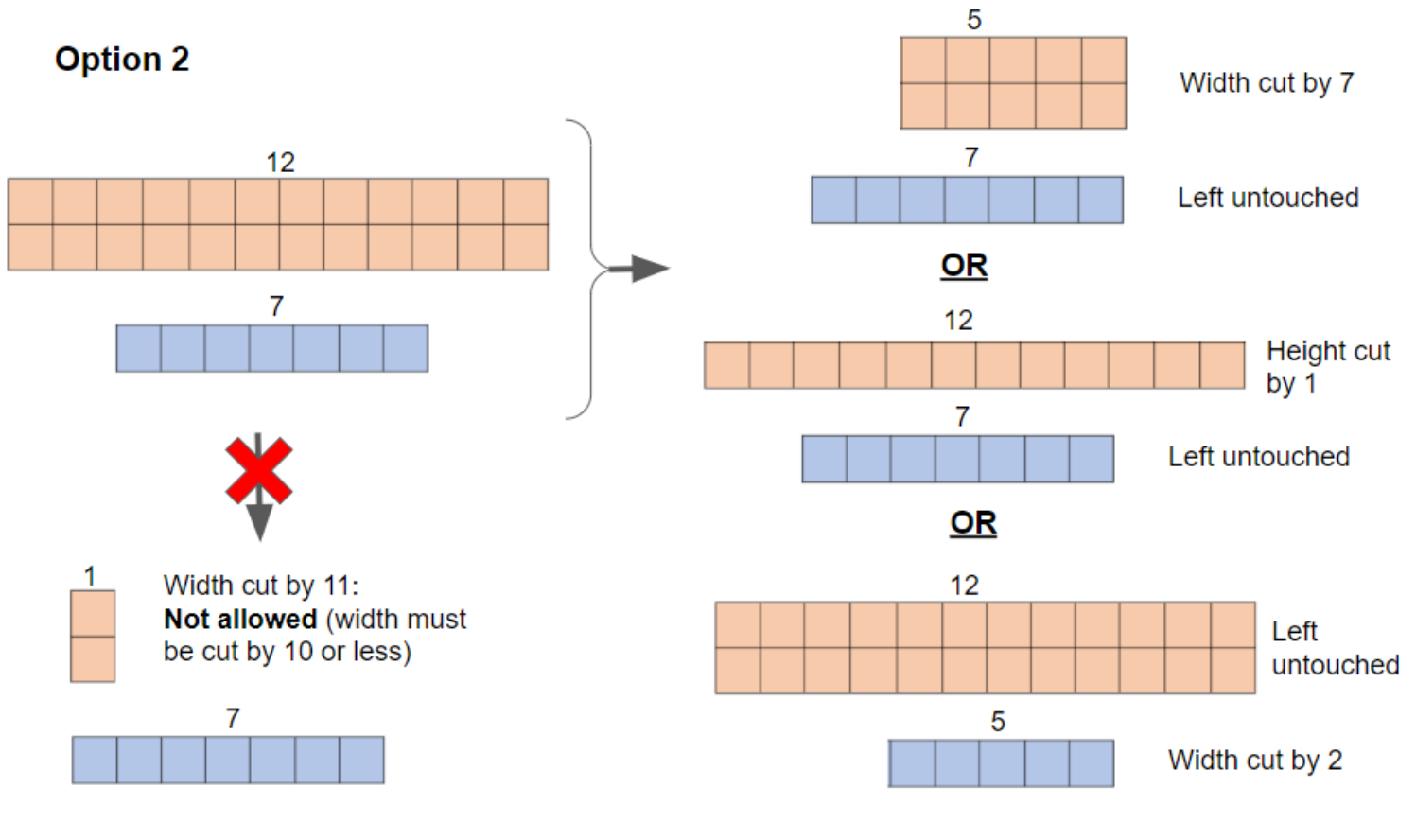
You move first. The player who leaves the other player with two 1×1 boards that can no longer be cut further is declared the winner.

You are deciding whether to place a bet against Tyler, so you want to know **whether it is possible for you to guarantee a win**, given the starting dimensions of the two boards.

NOTE FOR PYTHON USERS: If your program receives TLE (time limit exceeded), you should try submitting using the PyPy interpreter: when you are submitting your code, try using "PyPy 3" or "PyPy 2" as the language, instead of "Python 3" or "Python 2".

Input Specification

The input will be one line, containing four space-separated integers: h_1, w_1, h_2, and w_2, indicating that the two boards you start with are $h_1 \times w_1$ and $h_2 \times w_2$.

Output Specification

Output the character ⟨W⟩ if you can force a win, or ⟨L⟩ if your opponent Tyler can force a win.

Constraints and Partial Marks

For 2 of the 10 available marks, $h_1 = h_2 = 1$ and $w_1, w_2 \leq 10$.

For another 2 of the 10 available marks, $h_1 = h_2 = 1$ and $w_1, w_2 \leq 300$.

For another 2 of the 10 available marks, $1 \leq h_1, h_2 \leq 2$ and $w_1, w_2 \leq 10$.

For the remaining 4 marks, $1 \leq h_1, h_2 \leq 2$ and $w_1, w_2 \leq 300$.

Sample Input 1

```
2 2 1 3
```

Sample Output 1

```
W
```

Explanation for Sample Output 1

The sample input indicates that you start with a 2×2 board and a 1×3 board.

On your first turn, if you ignore the 2×2 board, and cut 2 off the width of the 1×3 board to leave a 1×1 board (this is permitted in Option 2), Tyler will get a 2×2 board and a 1×1 board on his turn.

It is easy to see that no matter how Tyler plays his move, he will leave you with at least one 1×2 board. As such, on your second turn, you just need to cut a 1×2 board into two 1×1 boards, and discard the other board (this is allowed in Option 1). Now, Tyler is left with two 1×1 boards that cannot be cut further, so he loses, and you win.

Since you are able to force a win, the output is ⟨W⟩.

Sample Input 2

```
2 2 1 1
```

Sample Output 2

L

Explanation for Sample Output 2

Similar to the example above, if you start with a 2×2 board and a 1×1 board, no matter what you do, you will always leave Tyler with at least one 1×2 board. Tyler can use Option 1 and cut the 1×2 board into two 1×1 boards and discard the other board, which is a loss for you.

Since you cannot force a win (instead, Tyler can force you to lose), the output is L.