A. Mister B and Boring Game

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

Sometimes Mister B has free evenings when he doesn't know what to do. Fortunately, Mister B found a new game, where the player can play against aliens.

All characters in this game are lowercase English letters. There are two players: Mister B and his competitor.

Initially the players have a string s consisting of the first a English letters in alphabetical order (for example, if $a = 5$, then s equals to "abcde").

The players take turns appending letters to string s. Mister B moves first.

Mister B must append exactly b letters on each his move. He can arbitrary choose these letters. His opponent adds exactly a letters on each move.

Mister B quickly understood that his opponent was just a computer that used a simple algorithm. The computer on each turn considers the suffix of string s of length a and generates a string t of length a such that all letters in the string t are distinct and don't appear in the considered suffix. From multiple variants of t lexicographically minimal is chosen (if $a = 4$ and the suffix is "bfdd", the computer chooses string t equal to "aceg"). After that the chosen string t is appended to the end of s.

Mister B soon found the game boring and came up with the following question: what can be the minimum possible number of different letters in string s on the segment between positions l and r, inclusive. Letters of string s are numerated starting from 1.

Input

First and only line contains four space-separated integers: a, b, l and r ($1 \le a, b \le 12$, $1 \le l \le r \le 10^9$) — the numbers of letters each player appends and the bounds of the segment.

Output

Print one integer — the minimum possible number of different letters in the segment from position l to position r, inclusive, in string s.

Examples

input
1 1 1 8
output
2

input
4 2 2 6
output
3

input
3 7 4 6
output
1

Note

In the first sample test one of optimal strategies generate string s = "`abababab...`", that's why answer is 2.

In the second sample test string s = "`abcdbcaefg...`" can be obtained, chosen segment will look like "`bcdbc`", that's why answer is 3.

In the third sample test string s = "`abczzzacad...`" can be obtained, chosen, segment will look like "`zzz`", that's why answer is 1.