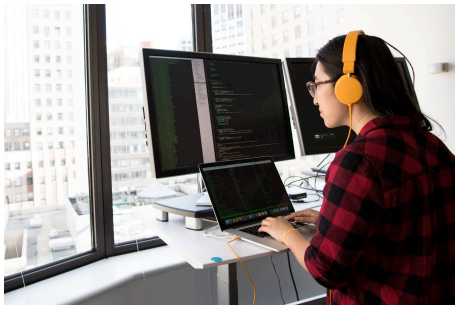
Encrypting Intervals

Time limit: 15 seconds



Gabriella has just made an incredible breakthrough in the world of encryption! Her new encryption algorithm allows communication between two people. Most of the details of this new encryption scheme are incomprehensible, but the crucial part of the encryption algorithm involves two intervals of positive integers.

The strength of the encryption algorithm is equal to the number of pairs of positive integers (one from each interval) that are **relatively prime** (that is, no positive integer other than 1 divides both numbers).
For example, if the two intervals are $[1, 2, 3, 4, 5]$ and $[3, 4, 5, 6]$, then the pairs (1,3), (1,4), (1,5), (1,6), (2,3),
(2,5), (3,4), (3,5), (4,3), (4,5), (5,3), (5,4), (5,6) are all relatively prime. Thus**, the strength is 13**.

Given the two positive intervals, what is the strength of the **encryption algorithm**?

Input

The first line of input consists of two integers, a and b ($1 \leq a \leq b \leq 10^4$ and $b - a \leq 10^4$), which are the two (inclusive) endpoints of the first interval.

The second line of input consists of two integers, c and d ($1 \leq c \leq d \leq 10^4$ and $d - c \leq 10^4$), which are the two (inclusive) endpoints of the second interval.

Output

Display the strength of the encryption algorithm.

Sample Input 1 **Sample Output 1**

1 5 3 6	13

Sample Input 2 **Sample Output 2**

10 50 51 100	1232

Submission guidelines: You need to submit three files (you don't need to zip them).

(1) Write one page document (upload pdf version of the doc) describing your algorithm or pseudocode. You should describe ***why and how*** your algorithm design should be ***efficient*** (the corresponding program should run fast).
(2) One program file – actual C/C++, Java, or Python code file.
(3) A screenshot of your program execution.

Make sure your code finishes its execution within 15 seconds for the largest possible input. You will only obtain a maximum of 80% of the score of your program runs slower than 15 seconds.

Hints are provided below. If you want to challenge yourself, don't read the hints first.

Hints: This problem is an application of ***prime factors***, ***Sieve of Eratosthenes*** and ***greatest common divisor*** **(gcd)**.