Between Two Sets ★

239 more points to get your next star!

Rank: 651206 | Points: 236/475

Problem Solving ★★★

Problem Submissions Leaderboard Editorial

There will be two arrays of integers. Determine all integers that satisfy the following two conditions:

1. The elements of the first array are all factors of the integer being considered
2. The integer being considered is a factor of all elements of the second array

These numbers are referred to as being between the two arrays. Determine how many such numbers exist.

Example

$a = [2, 6]$

$b = [24, 36]$

There are two numbers between the arrays: 6 and 12.

$6\%2 = 0$, $6\%6 = 0$, $24\%6 = 0$ and $36\%6 = 0$ for the first value.

$12\%2 = 0$, $12\%6 = 0$ and $24\%12 = 0$, $36\%12 = 0$ for the second value. Return 2.

Function Description

Complete the getTotalX function in the editor below. It should return the number of integers that are betwen the sets.

getTotalX has the following parameter(s):

- int a[n]: an array of integers
- int b[m]: an array of integers

Returns

- int: the number of integers that are between the sets

Input Format

The first line contains two space-separated integers, n and m, the number of elements in arrays a and b.

The second line contains n distinct space-separated integers $a[i]$ where $0 \leq i < n$.

The third line contains m distinct space-separated integers $b[j]$ where $0 \leq j < m$.

Constraints

- $1 \leq n, m \leq 10$
- $1 \leq a[i] \leq 100$
- $1 \leq b[j] \leq 100$

Sample Input

```
2 3
2 4
16 32 96
```

Sample Output

```
3
```

Explanation

2 and 4 divide evenly into 4, 8, 12 and 16.

4, 8 and 16 divide evenly into 16, 32, 96.

4, 8 and 16 are the only three numbers for which each element of a is a factor and each is a factor of all elements of b.

Change Theme Language Java 8

```java
import java.util.Scanner;

public class BetweenTwoSets {

  public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    int n = sc.nextInt();
    int m = sc.nextInt();
    int gcd = 0;

    int a[] = new int[n];
    int b[] = new int[m];

    for (int i = 0; i < n; i++) {
      a[i] = sc.nextInt();
    }
    int lcm = a[0];
    for (int i = 0; i < m; i++) {
      b[i] = sc.nextInt();
      gcd = findGCD(b[i], gcd);
    }
    for (int i = 0; i < n - 1; i++) {
      lcm = (lcm * a[i + 1]) / findGCD(a[i + 1], lcm);
    }
```

Line: 39 Col: 2

Upload Code as File Test against custom input Run Code Submit Code

Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Sample Test case 0

Sample Test case 1

Input (stdin) Download

```
2 3
2 4
16 32 96
```

Your Output (stdout)

```
3
```

Expected Output Download

```
3
```