You will be given two arrays of integers. You will be asked to 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. You must determine how many such numbers exist.

Function Description

Complete the function which is described by the below function signature.

```
integer getTotalX(integer_array a, integer_array b) {
    # Return the number of integers between the two sets
}
a: array of elements in set A
b: array of elements in set B
```

Constraints

- $1 \leq n, m \leq 10$
- $1 \leq a_i \leq 100$
- $1 \leq b_i \leq 100$

Raw Input Format

The first line contains two space-separated integers describing the respective values of n, the number of elements in array a, and m, the number of elements in array b.
The second line contains n distinct space-separated integers describing $a_0, a_1, \ldots, a_{n-1}$.
The third line contains m distinct space-separated integers describing $b_0, b_1, \ldots, b_{m-1}$.

Sample Input 0

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

Sample Output 0

```
3
```

Explanation 0

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.