 Practice · Compete · Jobs · Rank · Leaderboard

Dashboard › Algorithms › Dynamic Programming › GCD Matrix

Badge Progress (Details)

Points: 126   Rank: 270330

GCD Matrix 🔖

H by ma5termind

| Problem | Submissions | Leaderboard | Discussions | Editorial 🔒 |

Alex has two arrays defined as $A = [a_0, a_1, \ldots, a_{n-1}]$ and $B = [b_0, b_1, \ldots, b_{m-1}]$. He created an $n \times m$ matrix, M, where $M_{i,j} = \gcd(a_i, b_j)$ for each i, j in M. Recall that $\gcd(a, b)$ is the greatest common divisor of a and b.

For example, if $A = [2, 3]$ and $B = [5, 6]$, he builds $M = [[1, 2], [1, 3]]$ like so:

(i, j)	0	1
0	$\gcd(2, 5) = 1$	$\gcd(2, 6) = 2$
1	$\gcd(3, 5) = 1$	$\gcd(3, 6) = 3$

Alex's friend Kiara loves matrices, so he gives her q questions about matrix M where each question is in the form of some submatrix of M with its upper-left corner at M_{r_1, c_1} and its bottom-right corner at M_{r_2, c_2}. For each question, find and print the number of *distinct* integers in the given submatrix on a new line.

Input Format

The first line contains three space-separated integers describing the respective values of n (the size of array A), m (the size of array B), and q (Alex's number of questions).
The second line contains n space-separated integers describing $a_0, a_1, \ldots, a_{n-1}$.
The third line contains m space-separated integers describing $b_0, b_1, \ldots, b_{m-1}$.
Each line i of the q subsequent lines contains four space-separated integers describing the respective values of r_1, c_1, r_2, and c_2 for the ith question (i.e., defining a submatrix with upper-left corner (r_1, c_1) and bottom-right corner (r_2, c_2)).

Constraints

- $1 \leq n, m \leq 10^5$
- $1 \leq a_i, b_i \leq 10^5$
- $1 \leq q \leq 10$
- $0 \leq r_1, r_2 < n$
- $0 \leq c_1, c_2 < m$

Scoring

- $1 \leq n, m \leq 1000$ for 25% of score.
- $1 \leq n, m \leq 10^5$ for 100% of score.

Output Format

For each of Alex's questions, print the number of *distinct* integers in the given submatrix on a new line.

Sample Input 0

```
3 3 3
1 2 3
```

```
2 4 6
0 0 1 1
0 0 2 2
1 1 2 2
```

Sample Output 0

```
2
3
3
```

Explanation 0

Given $A = [1, 2, 3]$ and $B = [2, 4, 6]$, we build the following M:

(i, j)	0	1	2
0	$\gcd(1, 2) = 1$	$\gcd(1, 4) = 1$	$\gcd(1, 6) = 1$
1	$\gcd(2, 2) = 2$	$\gcd(2, 4) = 2$	$\gcd(2, 6) = 2$
2	$\gcd(3, 2) = 1$	$\gcd(3, 4) = 1$	$\gcd(3, 6) = 3$

The diagram below depicts the submatrices for each of the $q = 3$ questions in *green*:

Query 1   Query 2   Query 3

1. For the submatrix between $M_{0,0}$ and $M_{1,1}$, the set of integers is $\{1, 2\}$. The number of distinct integers is **2**.

2. For the submatrix between $M_{0,0}$ and $M_{2,2}$, the set of integers is $\{1, 2, 3\}$. The number of distinct integers is **3**.

3. For the submatrix between $M_{1,1}$ and $M_{2,2}$, the set of integers is $\{1, 2, 3\}$. The number of distinct integers is **3**.

f   y   in

Submissions: 122

Max Score: 60
Difficulty: Hard

Rate This Challenge:
☆ ☆ ☆ ☆ ☆

More

Current Buffer (saved locally, editable)	Java 7

```java
1  import java.io.*;
2  import java.util.*;
3  import java.text.*;
4  import java.math.*;
5  import java.util.regex.*;
6
7  public class Solution {
8
9      public static void main(String[] args) {
10         Scanner in = new Scanner(System.in);
11         int n = in.nextInt();
12         int m = in.nextInt();
13         int q = in.nextInt();
14         int[] a = new int[n];
15         for(int a_i=0; a_i < n; a_i++){
16             a[a_i] = in.nextInt();
17         }
18         int[] b = new int[m];
```

```java
19 ▼          for(int b_i=0; b_i < m; b_i++){
20 ▼              b[b_i] = in.nextInt();
21            }
22 ▼          for(int a0 = 0; a0 < q; a0++){
23                int r1 = in.nextInt();
24                int c1 = in.nextInt();
25                int r2 = in.nextInt();
26                int c2 = in.nextInt();
27                // your code goes here
28            }
29        }
30 }
31
```

Line: 1 Col: 1

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

---

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature