Problem 3312: Sorted GCD Pair Queries

Problem Information

Difficulty: Hard
Acceptance Rate: 21.20%
Paid Only: No
Tags: Array, Hash Table, Math, Binary Search, Combinatorics, Counting, Number Theory, Prefix Sum

Problem Description

You are given an integer array `nums` of length `n` and an integer array `queries`.

Let `gcdPairs` denote an array obtained by calculating the GCD of all possible pairs `(nums[i], nums[j])`, where `0 <= i < j < n`, and then sorting these values in **ascending** order.

For each query `queries[i]`, you need to find the element at index `queries[i]` in `gcdPairs`.

Return an integer array `answer`, where `answer[i]` is the value at `gcdPairs[queries[i]]` for each query.

The term `gcd(a, b)` denotes the **greatest common divisor** of `a` and `b`.

Example 1:

Input: nums = [2,3,4], queries = [0,2,2]

Output: [1,2,2]

Explanation:

`gcdPairs = [gcd(nums[0], nums[1]), gcd(nums[0], nums[2]), gcd(nums[1], nums[2])] = [1, 2, 1]`.

After sorting in ascending order, `gcdPairs = [1, 1, 2]`.

So, the answer is `[gcdPairs[queries[0]], gcdPairs[queries[1]], gcdPairs[queries[2]]] = [1, 2, 2]`.

Example 2:

Input: nums = [4,4,2,1], queries = [5,3,1,0]

Output: [4,2,1,1]

Explanation:

`gcdPairs` sorted in ascending order is `[1, 1, 1, 2, 2, 4]`.

Example 3:

Input: nums = [2,2], queries = [0,0]

Output: [2,2]

Explanation:

`gcdPairs = [2]`.

Constraints:

* `2 <= n == nums.length <= 105` * `1 <= nums[i] <= 5 * 104` * `1 <= queries.length <= 105` * `0 <= queries[i] < n * (n - 1) / 2`

Code Snippets

C++:

```
class Solution {
public:
vector<int> gcdValues(vector<int>& nums, vector<long long>& queries) {

}
};
```

Java:

```
class Solution {
public int[] gcdValues(int[] nums, long[] queries) {

}
}
```

Python3:

```
class Solution:
def gcdValues(self, nums: List[int], queries: List[int]) -> List[int]:
```