

Problem 1714: Sum Of Special Evenly-Spaced Elements In Array

Problem Information

Difficulty: Hard

Acceptance Rate: 49.69%

Paid Only: Yes

Tags: Array, Dynamic Programming

Problem Description

You are given a **0-indexed** integer array `nums` consisting of `n` non-negative integers.

You are also given an array `queries`, where `queries[i] = [xi, yi]`. The answer to the `i`th query is the sum of all `nums[j]` where `xi ≤ j < n` and `(j - xi)` is divisible by `yi`.

Return an array `answer` where `answer.length == queries.length` and `answer[i]` is the answer to the `i`th query **modulo** `109 + 7`.

Example 1:

Input: `nums = [0,1,2,3,4,5,6,7]`, `queries = [[0,3],[5,1],[4,2]]` **Output:** `[9,18,10]`

Explanation: The answers of the queries are as follows: 1) The `j` indices that satisfy this query are 0, 3, and 6. `nums[0] + nums[3] + nums[6] = 9` 2) The `j` indices that satisfy this query are 5, 6, and 7. `nums[5] + nums[6] + nums[7] = 18` 3) The `j` indices that satisfy this query are 4 and 6. `nums[4] + nums[6] = 10`

Example 2:

Input: `nums = [100,200,101,201,102,202,103,203]`, `queries = [[0,7]]` **Output:** `[303]`

Constraints:

`n == nums.length` `1 ≤ n ≤ 5 * 104` `0 ≤ nums[i] ≤ 109` `1 ≤ queries.length ≤ 1.5 * 105` `0 ≤ xi < n` `1 ≤ yi ≤ 5 * 104`

Code Snippets

C++:

```
class Solution {  
public:  
    vector<int> solve(vector<int>& nums, vector<vector<int>>& queries) {  
  
    }  
};
```

Java:

```
class Solution {  
    public int[] solve(int[] nums, int[][] queries) {  
  
    }  
}
```

Python3:

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