6466. Number of Beautiful Pairs

My Submissions (/contest/weekly-contest-351/problems/number-of-beautiful-pairs/submissions/) Back to Contest (/contest/weekly-contest-351/)

You are given a **0-indexed** integer array `nums`. A pair of indices `i, j` where `0 <= i < j < nums.length` is called beautiful if the **first digit** of `nums[i]` and the **last digit** of `nums[j]` are **coprime**.

Return *the total number of beautiful pairs in* `nums`.

Two integers `x` and `y` are **coprime** if there is no integer greater than 1 that divides both of them. In other words, `x` and `y` are coprime if `gcd(x, y) == 1`, where `gcd(x, y)` is the **greatest common divisor** of `x` and `y`.

User Accepted:	3790
User Tried:	5697
Total Accepted:	3798
Total Submissions:	6949
Difficulty:	Easy

Example 1:

```
Input: nums = [2,5,1,4]
Output: 5
Explanation: There are 5 beautiful pairs in nums:
When i = 0 and j = 1: the first digit of nums[0] is 2, and the last digit of nums[1] is 5. We can confirm that 2 and 5 are cop
When i = 0 and j = 2: the first digit of nums[0] is 2, and the last digit of nums[2] is 1. Indeed, gcd(2,1) == 1.
When i = 1 and j = 2: the first digit of nums[1] is 5, and the last digit of nums[2] is 1. Indeed, gcd(5,1) == 1.
When i = 1 and j = 3: the first digit of nums[1] is 5, and the last digit of nums[3] is 4. Indeed, gcd(5,4) == 1.
When i = 2 and j = 3: the first digit of nums[2] is 1, and the last digit of nums[3] is 4. Indeed, gcd(1,4) == 1.
Thus, we return 5.
```

Example 2:

```
Input: nums = [11,21,12]
Output: 2
Explanation: There are 2 beautiful pairs:
When i = 0 and j = 1: the first digit of nums[0] is 1, and the last digit of nums[1] is 1. Indeed, gcd(1,1) == 1.
When i = 0 and j = 2: the first digit of nums[0] is 1, and the last digit of nums[2] is 2. Indeed, gcd(1,2) == 1.
Thus, we return 2.
```

Constraints:

- `2 <= nums.length <= 100`
- `1 <= nums[i] <= 9999`
- `nums[i] % 10 != 0`

JavaScript

```javascript
const gcd = (a, b) => b == 0 ? a : gcd(b, a % b);

const countBeautifulPairs = (a) => {
    let n = a.length, res = 0;
    for (let i = 0; i < n; i++) {
        for (let j = i + 1; j < n; j++) {
            if (ok(a[i], a[j])) res++;
        }
    }
    return res;
};

const ok = (x, y) => {
    let sx = x + '', sy = y + '';
    return gcd(sx[0] - '0', sy[sy.length - 1] - '0') == 1;
};
```

Custom Testcase Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/978924517/) More Details (/submissions/detail/978924517/)

Share your acceptance!