Problem 1814: Count Nice Pairs in an Array

Problem Information

Difficulty: Medium
Acceptance Rate: 48.49%
Paid Only: No
Tags: Array, Hash Table, Math, Counting

Problem Description

You are given an array `nums` that consists of non-negative integers. Let us define `rev(x)` as the reverse of the non-negative integer `x`. For example, `rev(123) = 321`, and `rev(120) = 21`. A pair of indices `(i, j)` is **nice** if it satisfies all of the following conditions:

* `0 <= i < j < nums.length` * `nums[i] + rev(nums[j]) == nums[j] + rev(nums[i])`

Return _the number of nice pairs of indices_. Since that number can be too large, return it **modulo** `109 + 7`.

Example 1:

Input: nums = [42,11,1,97] **Output:** 2 **Explanation:** The two pairs are: - (0,3) : 42 + rev(97) = 42 + 79 = 121, 97 + rev(42) = 97 + 24 = 121. - (1,2) : 11 + rev(1) = 11 + 1 = 12, 1 + rev(11) = 1 + 11 = 12.

Example 2:

Input: nums = [13,10,35,24,76] **Output:** 4

Constraints:

* `1 <= nums.length <= 105` * `0 <= nums[i] <= 109`

Code Snippets

C++:

```
class Solution {
public:
int countNicePairs(vector<int>& nums) {

}
};
```

Java:

```
class Solution {
public int countNicePairs(int[] nums) {

}
}
```

Python3:

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