Problem 2354: Number of Excellent Pairs

Problem Information

Difficulty: Hard
Acceptance Rate: 48.65%
Paid Only: No
Tags: Array, Hash Table, Binary Search, Bit Manipulation

Problem Description

You are given a **0-indexed** positive integer array `nums` and a positive integer `k`.

A pair of numbers `(num1, num2)` is called **excellent** if the following conditions are satisfied:

* **Both** the numbers `num1` and `num2` exist in the array `nums`. * The sum of the number of set bits in `num1 OR num2` and `num1 AND num2` is greater than or equal to `k`, where `OR` is the bitwise **OR** operation and `AND` is the bitwise **AND** operation.

Return _the number of**distinct** excellent pairs_.

Two pairs `(a, b)` and `(c, d)` are considered distinct if either `a != c` or `b != d`. For example, `(1, 2)` and `(2, 1)` are distinct.

Note that a pair `(num1, num2)` such that `num1 == num2` can also be excellent if you have at least **one** occurrence of `num1` in the array.

Example 1:

Input: nums = [1,2,3,1], k = 3 **Output:** 5 **Explanation:** The excellent pairs are the following: - (3, 3). (3 AND 3) and (3 OR 3) are both equal to (11) in binary. The total number of set bits is 2 + 2 = 4, which is greater than or equal to k = 3. - (2, 3) and (3, 2). (2 AND 3) is equal to (10) in binary, and (2 OR 3) is equal to (11) in binary. The total number of set bits is 1 + 2 = 3. - (1, 3) and (3, 1). (1 AND 3) is equal to (01) in binary, and (1 OR 3) is equal to (11) in binary. The total number of set bits is 1 + 2 = 3. So the number of excellent pairs is 5.

Example 2:

Input: nums = [5,1,1], k = 10 **Output:** 0 **Explanation:** There are no excellent pairs for this array.

Constraints:

* `1 <= nums.length <= 105` * `1 <= nums[i] <= 109` * `1 <= k <= 60`

Code Snippets

C++:

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

}
};
```

Java:

```
class Solution {
public long countExcellentPairs(int[] nums, int k) {

}
}
```

Python3:

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