Problem 2206: Divide Array Into Equal Pairs

Problem Information

Difficulty: Easy
Acceptance Rate: 79.26%
Paid Only: No
Tags: Array, Hash Table, Bit Manipulation, Counting

Problem Description

You are given an integer array `nums` consisting of `2 * n` integers.

You need to divide `nums` into `n` pairs such that:

* Each element belongs to **exactly one** pair. * The elements present in a pair are **equal**.

Return `true` _if nums can be divided into_ `n` _pairs, otherwise return_ `false`.

Example 1:

Input: nums = [3,2,3,2,2,2] **Output:** true **Explanation:** There are 6 elements in nums, so they should be divided into 6 / 2 = 3 pairs. If nums is divided into the pairs (2, 2), (3, 3), and (2, 2), it will satisfy all the conditions.

Example 2:

Input: nums = [1,2,3,4] **Output:** false **Explanation:** There is no way to divide nums into 4 / 2 = 2 pairs such that the pairs satisfy every condition.

Constraints:

* `nums.length == 2 * n` * `1 <= n <= 500` * `1 <= nums[i] <= 500`

Code Snippets

C++:

```cpp
class Solution {
public:
bool divideArray(vector<int>& nums) {


}
};
```

Java:

```java
class Solution {
public boolean divideArray(int[] nums) {


}
}
```

Python3:

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