Problem 442: Find All Duplicates in an Array

Problem Information

Difficulty: Medium
Acceptance Rate: 76.66%
Paid Only: No
Tags: Array, Hash Table, Sorting

Problem Description

Given an integer array `nums` of length `n` where all the integers of `nums` are in the range `[1, n]` and each integer appears **at most** **twice** , return _an array of all the integers that appears**twice**_.

You must write an algorithm that runs in `O(n)` time and uses only _constant_ auxiliary space, excluding the space needed to store the output

Example 1:

Input: nums = [4,3,2,7,8,2,3,1] **Output:** [2,3]

Example 2:

Input: nums = [1,1,2] **Output:** [1]

Example 3:

Input: nums = [1] **Output:** []

Constraints:

* `n == nums.length` * `1 <= n <= 105` * `1 <= nums[i] <= n` * Each element in `nums` appears **once** or **twice**.

Code Snippets

C++:

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

}
};
```

Java:

```
class Solution {
public List<Integer> findDuplicates(int[] nums) {

}
}
```

Python3:

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