Problem 448: Find All Numbers Disappeared in an Array

Problem Information

Difficulty: Easy
Acceptance Rate: 63.11%
Paid Only: No
Tags: Array, Hash Table

Problem Description

Given an array `nums` of `n` integers where `nums[i]` is in the range `[1, n]`, return _an array of all the integers in the range_ `[1, n]` _that do not appear in_ `nums`.

Example 1:

Input: nums = [4,3,2,7,8,2,3,1] **Output:** [5,6]

Example 2:

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

Constraints:

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

Follow up: Could you do it without extra space and in `O(n)` runtime? You may assume the returned list does not count as extra space.

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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