Find all Duplicate Numbers (easy)

We'll cover the following

Problem Statement

We are given an unsorted array containing 'n' numbers taken from the range 1 to 'n'. The array has some numbers appearing twice, **find all these duplicate numbers without using any extra space**.

Example 1:

```
Input: [3, 4, 4, 5, 5]
Output: [4, 5]
```

Example 2:

```
Input: [5, 4, 7, 2, 3, 5, 3]
Output: [3, 5]
```

Try it yourself

Try solving this question here:

Java | Python3 | JS | C++

```java
import java.util.*;

class FindAllDuplicate {

  public static List<Integer> findNumbers(int[] nums) {
    List<Integer> duplicateNumbers = new ArrayList<>();
    // TODO: Write your code here
    return duplicateNumbers;
  }
}

```

Test | Save | Reset

Solution

This problem follows the **Cyclic Sort** pattern and shares similarities with Find the Duplicate Number. Following a similar approach, we will place each number at its correct index. After that, we will iterate through the array to find all numbers that are not at the correct indices. All these numbers are duplicates.

Code

Here is what our algorithm will look like:

Java | Python3 | C++ | JS

```java
import java.util.*;

class FindAllDuplicate {

  public static List<Integer> findNumbers(int[] nums) {
    int i = 0;
    while (i < nums.length) {
      if (nums[i] != nums[nums[i] - 1])
        swap(nums, i, nums[i] - 1);
      else
        i++;
    }
```

```java

    List<Integer> duplicateNumbers = new ArrayList<>();
    for (i = 0; i < nums.length; i++) {
      if (nums[i] != i + 1)
        duplicateNumbers.add(nums[i]);
    }

    return duplicateNumbers;
  }

  private static void swap(int[] arr, int i, int j) {
    int temp = arr[i];
    arr[i] = arr[j];
    arr[j] = temp;
  }

```

Run Save Reset

Time complexity

The time complexity of the above algorithm is $O(n)$.

Space complexity

Ignoring the space required for storing the duplicates, the algorithm runs in constant space $O(1)$.

Back
Find the Duplicate Number (easy)

Next
Problem Challenge 1

Mark as Completed

Report an Issue Ask a Question