

Find all Duplicate Numbers (easy)

We'll cover the following ^

- Problem Statement
- Try it yourself
- Solution
- Code
 - Time complexity
 - Space complexity

Problem Statement

We are given an unsorted array containing 'n' numbers taken from the range 1 to 'n'. The array has some duplicates, **find all the 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++

```
1 def find_all_duplicates(nums):
2     duplicateNumbers = []
3     # TODO: Write your code here
4     return duplicateNumbers
5
```

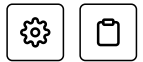


Solution

This problem follows the **Cyclic Sort** pattern and shares similarities with Find the Duplicate Number

(<https://www.educative.io/collection/page/5668639101419520/5671464854355968/4522012447866880/>). 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 JS

```
1 def find_all_duplicates(nums):
2     i = 0
3     while i < len(nums):
4         j = nums[i] - 1
5         if nums[i] != nums[j]:
6             nums[i], nums[j] = nums[j], nums[i] # swap
7         else:
8             i += 1
9
10    duplicateNumbers = []
11    for i in range(len(nums)):
12        if nums[i] != i + 1:
13            duplicateNumbers.append(nums[i])
14
15    return duplicateNumbers
16
17
18 def main():
19     print(find_all_duplicates([3, 4, 4, 5, 5]))
20     print(find_all_duplicates([5, 4, 7, 2, 3, 5, 3]))
21
22
23 main()
24
```

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

Next →

Find the Duplicate Number (easy)

Problem Challenge 1

☒ Mark as Completed

Report an Issue

Ask a Question

(https://discuss.educative.io/tag/find-all-duplicate-numbers-easy__pattern-cyclic-sort__grokking-the-coding-interview-patterns-for-coding-questions)

