



Find The Duplicate Number

Try to solve the Find The Duplicate Number problem.

We'll cover the following



- Statement
- Examples
- Understand the problem
- Figure it out!
- Try it yourself

Statement

Given an unsorted array of positive numbers, `nums`, such that the values lie in the range $[1, n]$, inclusive, and that there are $n + 1$ numbers in the array, find and return the duplicate number present in `nums`. There is only one repeated number in `nums`.

Note: You cannot modify the given array `nums`. You have to solve the problem using only constant extra space.

Constraints:

- $1 \leq n \leq 10^5$
- `nums.length` = $n + 1$
- $1 \leq \text{nums}[i] \leq n$
- All the integers in `nums` are unique, except for one integer that will appear more than once.



Examples

Sample example 1

Input

1	3	3	4	2	5
---	---	---	---	---	---

Output

Output	3
--------	---

1 of 5



Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

Find The Duplicate Number

1

What is the output if the following list of integers is given as input?

[1, 5, 4, 3, 2, 4, 6]

A) 5

B) 3

?

Tr



C) 4

D) 7

Submit Answer



Question 1 of 4
0 attempted



Reset Quiz ↻

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.



Drag and drop the cards to rearrange them in the correct sequence.

Return the **fast** pointer.

After the pointers meet,
traverse in **nums** again.

Move the **slow** pointer from
the start of **nums** and the **fast**
pointer from the meeting
point at the same speed (one
step) until they meet again.



Move pointers until they meet. The **slow** pointer moves once and the **fast** pointer moves twice as fast as the **slow** pointer.

Traverse in **nums** using the **fast** and **slow** pointers.



Try it yourself

Implement your solution in the following coding playground:

Java

usercode > FindDuplicate.java

```
1 import java.util.*;
2
3 public class FindDuplicate{
4     public static int findDuplicate(int[] nums) {
5
6         // Write your code here
7
8         return 0;
9     }
10 }
```

Powered by AI ?

Test Cases Results

Case 1

Case 2

Case 3

Tr

Input #1

[3,4,4,4,2]

Find The Duplicate Number

← Back

Solution: Circular Arra...

Next →

Solution: Find The Du...



Mark as
Completed

