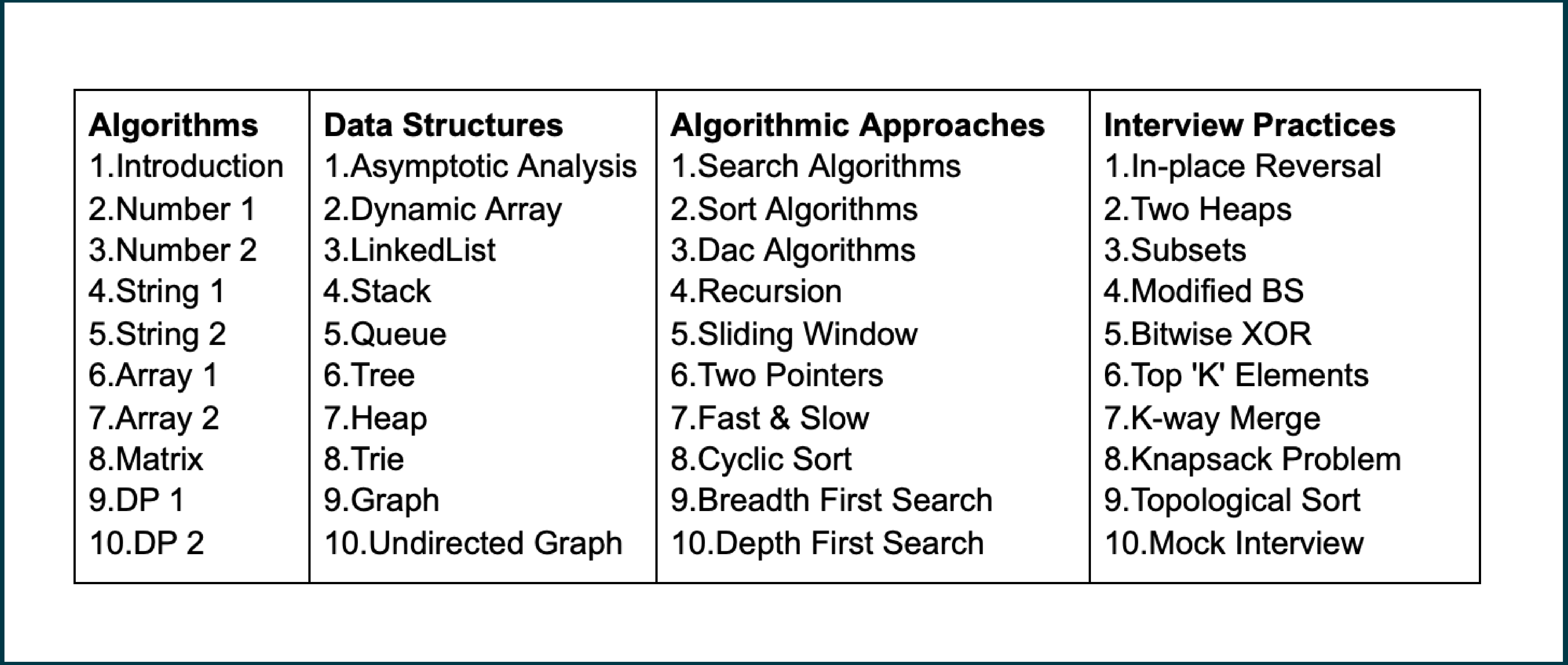
DSA - Algorithms

Array 1

Course Planning

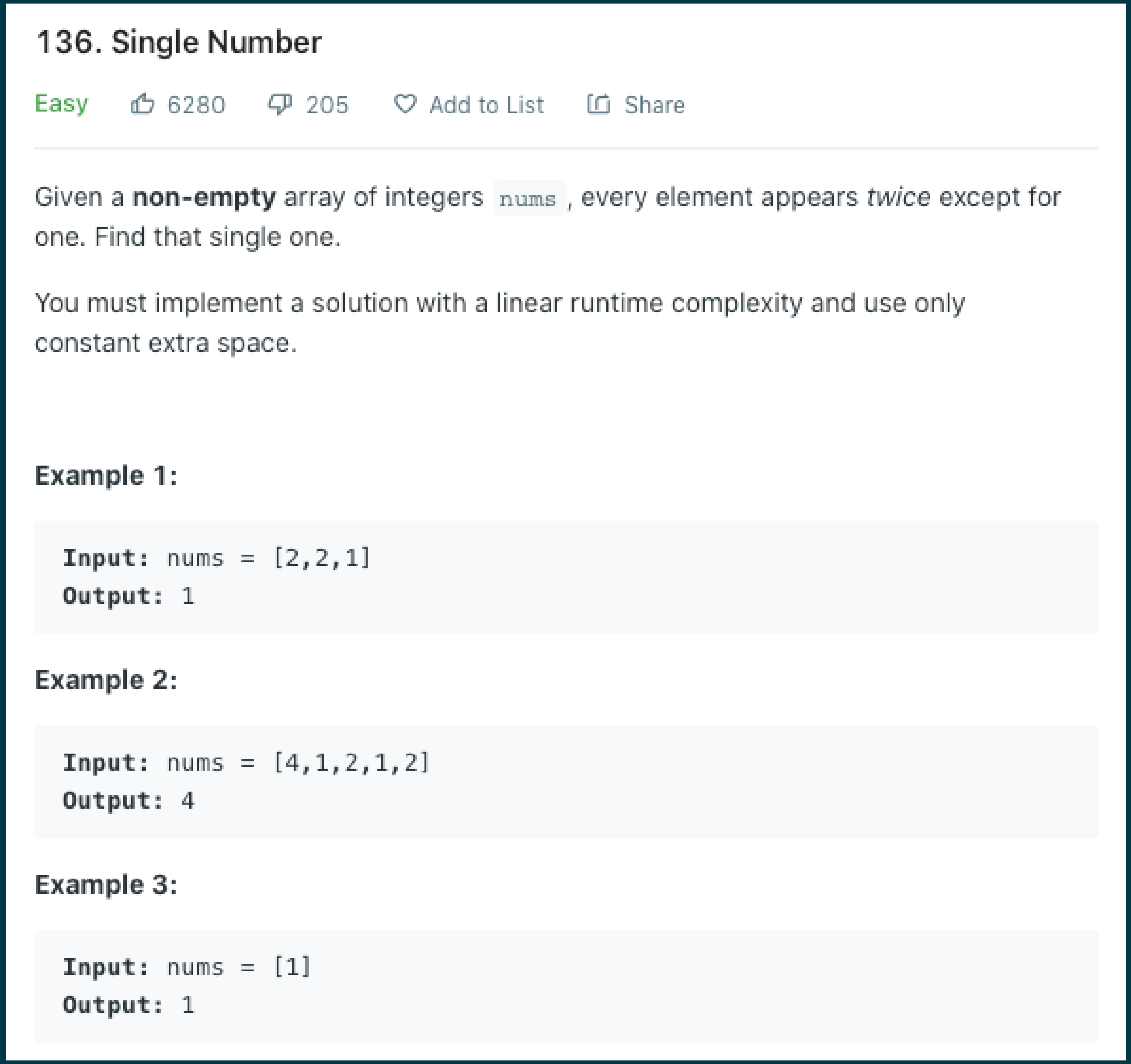


Algorithms	Data Structures	Algorithmic Approaches	Interview Practices
1.Introduction	1.Asymptotic Analysis	1.Search Algorithms	1.In-place Reversal
2.Number 1	2.Dynamic Array	2.Sort Algorithms	2.Two Heaps
3.Number 2	3.LinkedList	3.Dac Algorithms	3.Subsets
4.String 1	4.Stack	4.Recursion	4.Modified BS
5.String 2	5.Queue	5.Sliding Window	5.Bitwise XOR
6.Array 1	6.Tree	6.Two Pointers	6.Top 'K' Elements
7.Array 2	7.Heap	7.Fast & Slow	7.K-way Merge
8.Matrix	8.Trie	8.Cyclic Sort	8.Knapsack Problem
9.DP 1	9.Graph	9.Breadth First Search	9.Topological Sort
10.DP 2	10.Undirected Graph	10.Depth First Search	10.Mock Interview

https://kurbanovxurshid.medium.com/data-structures-and-algorithms-course-plan-1ab912ec1e17

Asked by Amazon

Explanation



136. Single Number

Given a **non-empty** array of integers `nums`, every element appears *twice* except for one. Find that single one.

You must implement a solution with a linear runtime complexity and use only constant extra space.

Example 1:

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

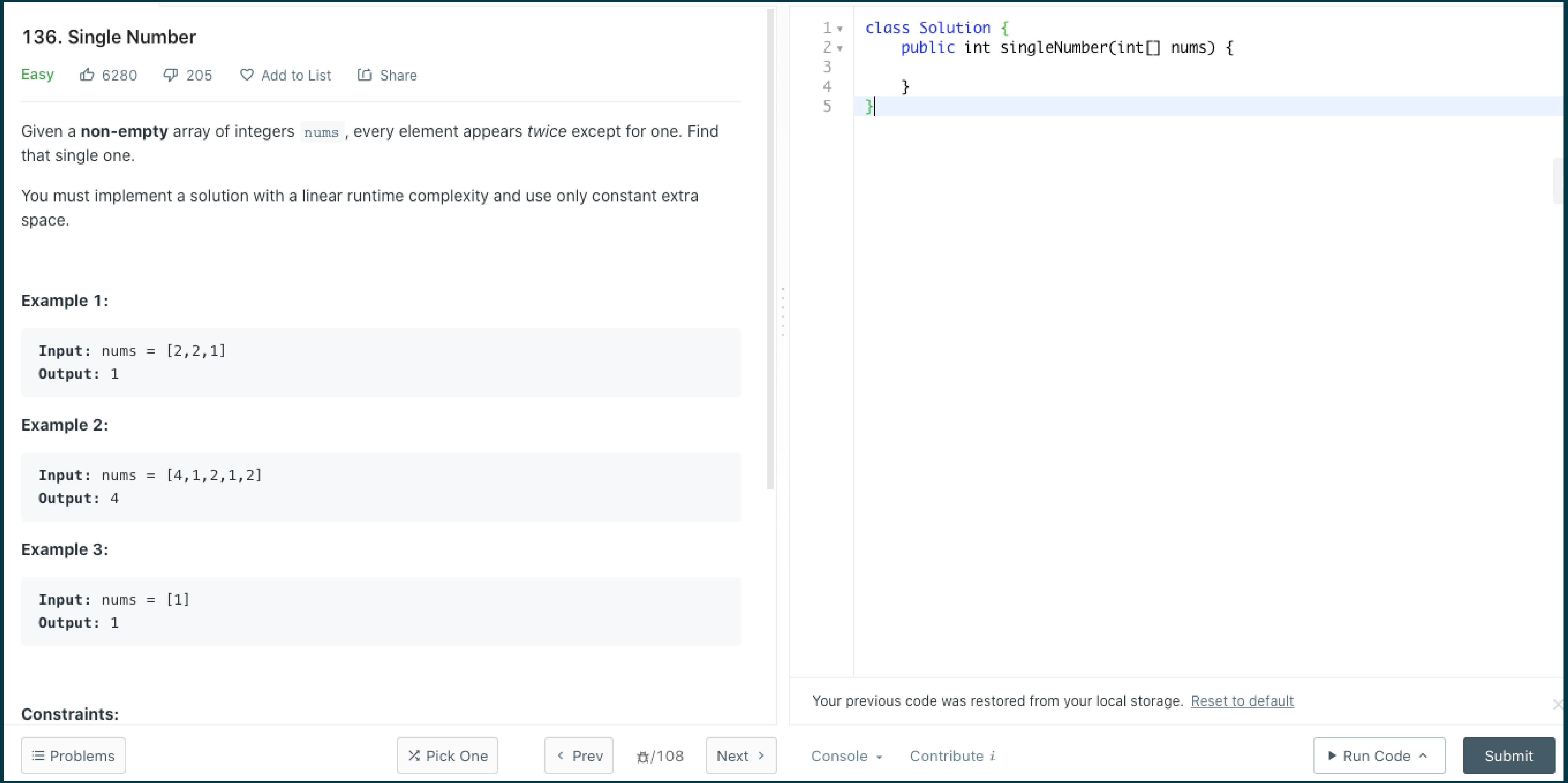
Example 2:

```
Input: nums = [4,1,2,1,2]
Output: 4
```

Example 3:

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

Single Number



136. Single Number

Easy 6280 205 Add to List Share

Given a **non-empty** array of integers `nums`, every element appears *twice* except for one. Find that single one.

You must implement a solution with a linear runtime complexity and use only constant extra space.

Example 1:

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

Example 2:

```
Input: nums = [4,1,2,1,2]
Output: 4
```

Example 3:

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

Constraints:

```
class Solution {
    public int singleNumber(int[] nums) {

    }
}
```

Your previous code was restored from your local storage. Reset to default

https://leetcode.com/problems/single-number/

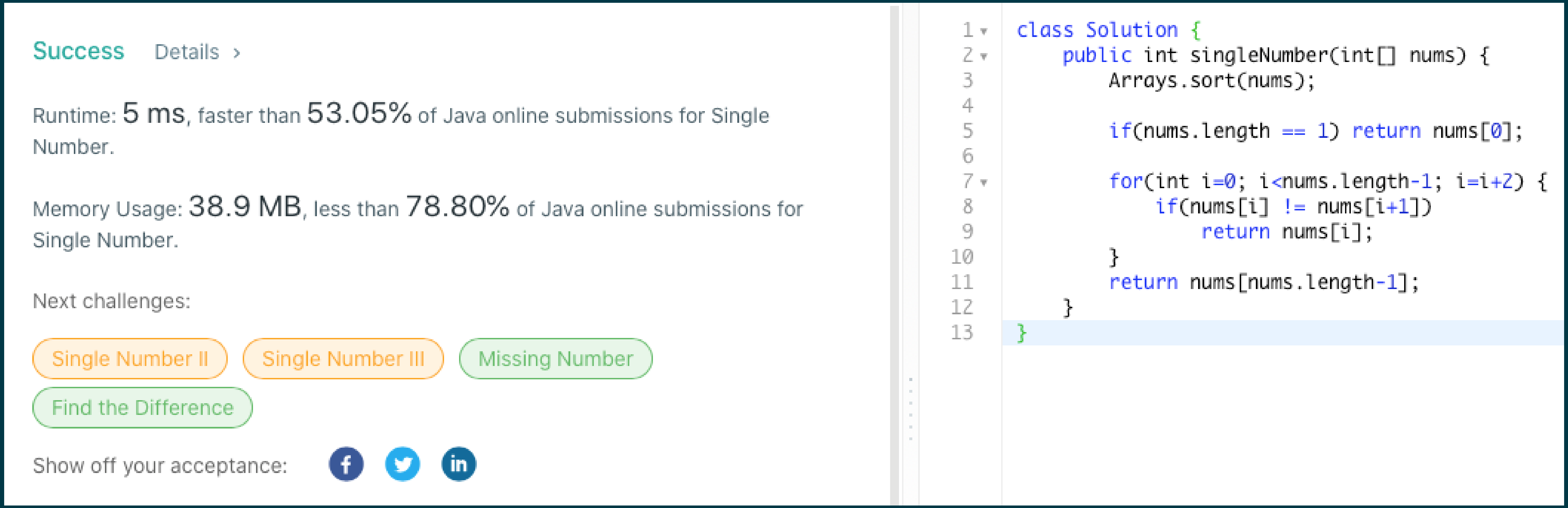
First Theory

[4,1,2,1,2,4,3]

Sort Array

[1,1,2,2,3,4,4]

First Solution


Success Details ›

Runtime: 5 ms, faster than 53.05% of Java online submissions for Single Number.

Memory Usage: 38.9 MB, less than 78.80% of Java online submissions for Single Number.

Next challenges:

Single Number II Single Number III Missing Number Find the Difference

Show off your acceptance:

```java
class Solution {
    public int singleNumber(int[] nums) {
        Arrays.sort(nums);

        if(nums.length == 1) return nums[0];

        for(int i=0; i<nums.length-1; i=i+2) {
            if(nums[i] != nums[i+1])
                return nums[i];
        }
        return nums[nums.length-1];
    }
}
```

Second Theory

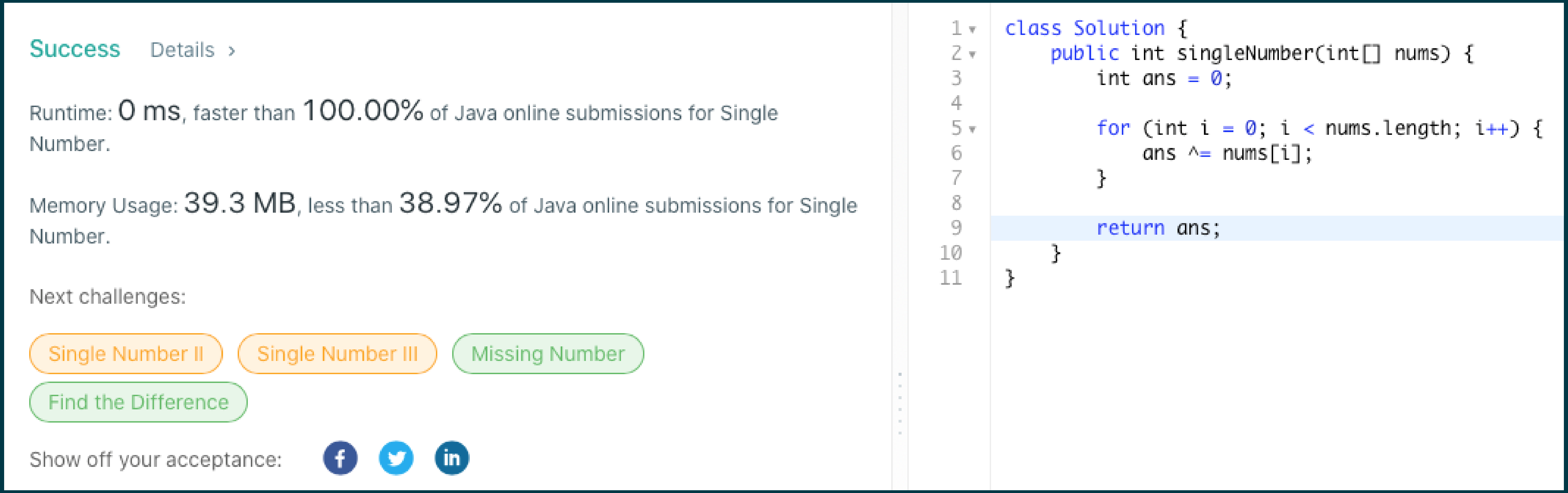
A xor A = 0

A xor B xor A = A xor A xor B = 0 xor B = B

[4,1,2,1,2,4,3]

1 xor 1 xor 2 xor 2 xor 3 xor 4 xor 4 = 0 xor 0 xor 3 xor 0 = 0 xor 3 = 3

Second Solution



Success Details ›

Runtime: 0 ms, faster than 100.00% of Java online submissions for Single Number.

Memory Usage: 39.3 MB, less than 38.97% of Java online submissions for Single Number.

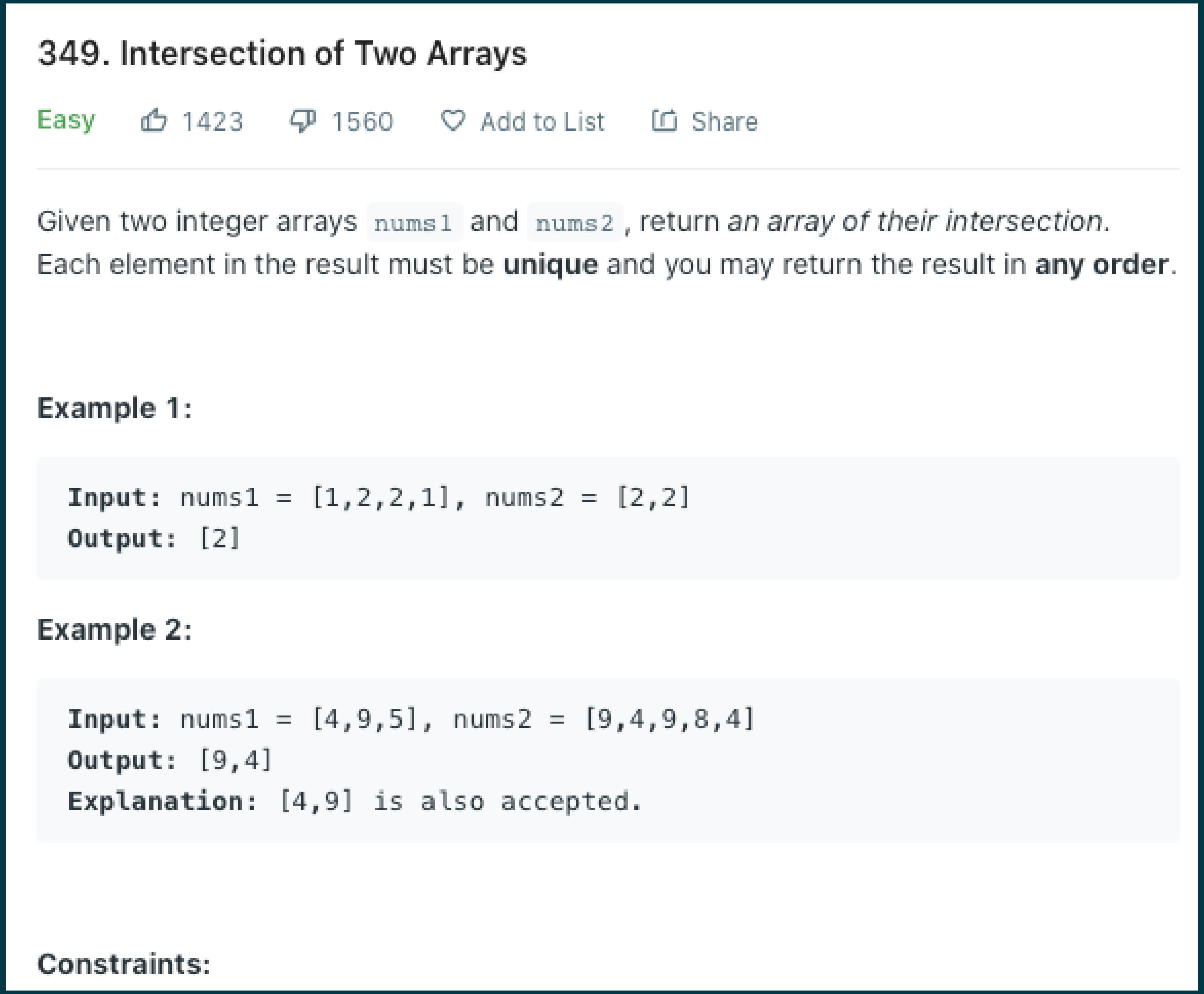
Next challenges:

```java
class Solution {
    public int singleNumber(int[] nums) {
        int ans = 0;

        for (int i = 0; i < nums.length; i++) {
            ans ^= nums[i];
        }

        return ans;
    }
}
```

Task 1 – Intersection of Two Arrays



349. Intersection of Two Arrays

Easy 1423 1560 Add to List Share

Given two integer arrays `nums1` and `nums2`, return *an array of their intersection*. Each element in the result must be **unique** and you may return the result in **any order**.

Example 1:

```
Input: nums1 = [1,2,2,1], nums2 = [2,2]
Output: [2]
```

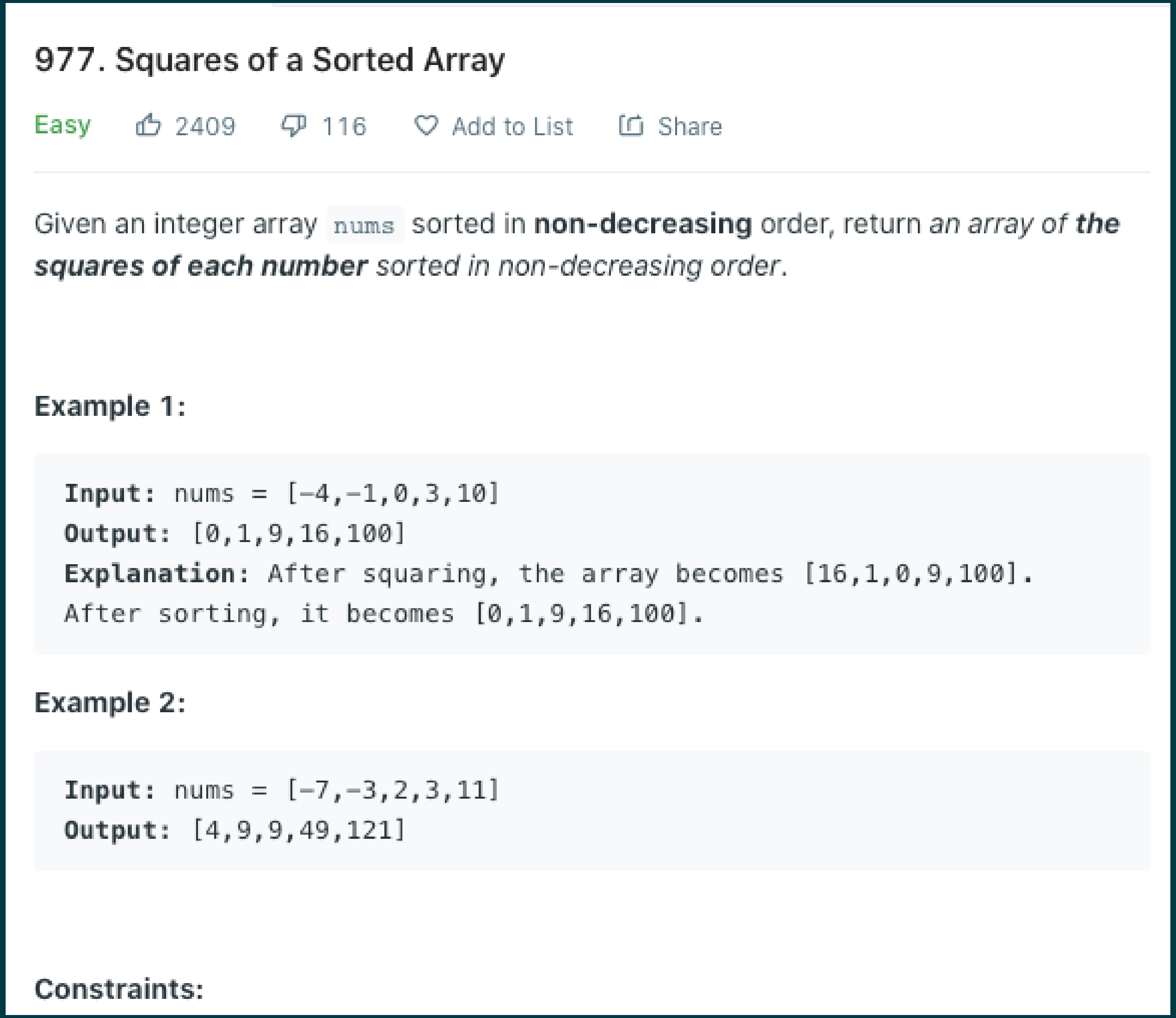
Example 2:

```
Input: nums1 = [4,9,5], nums2 = [9,4,9,8,4]
Output: [9,4]
Explanation: [4,9] is also accepted.
```

Constraints:

https://leetcode.com/problems/intersection-of-two-arrays/

Task 2 – Squares of a Sorted Array



977. Squares of a Sorted Array

Easy 2409 116 Add to List Share

Given an integer array `nums` sorted in **non-decreasing** order, return *an array of **the squares of each number** sorted in non-decreasing order*.

Example 1:

```
Input: nums = [-4,-1,0,3,10]
Output: [0,1,9,16,100]
Explanation: After squaring, the array becomes [16,1,0,9,100].
After sorting, it becomes [0,1,9,16,100].
```

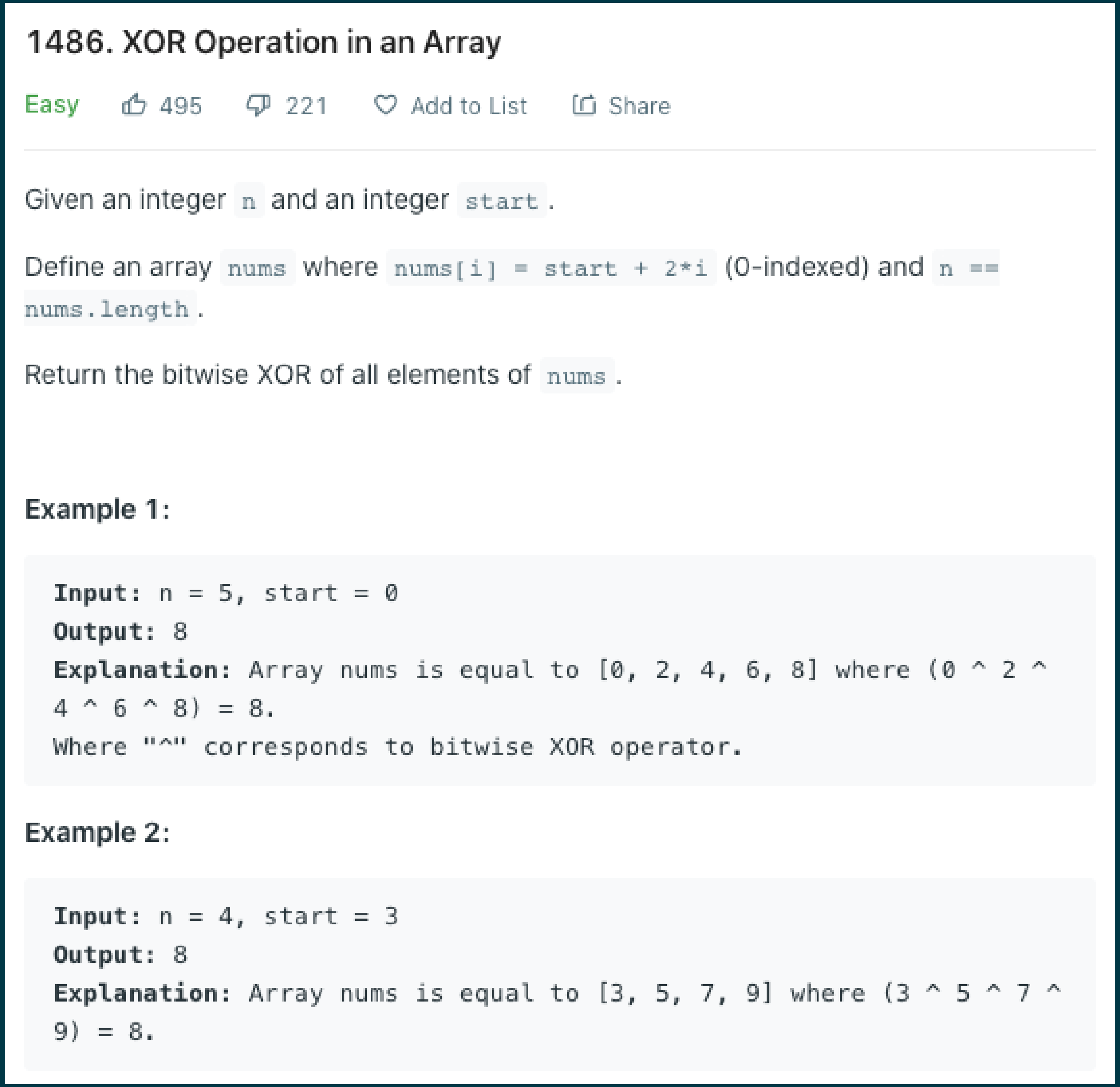
Example 2:

```
Input: nums = [-7,-3,2,3,11]
Output: [4,9,9,49,121]
```

Constraints:

https://leetcode.com/problems/squares-of-a-sorted-array/

Task 3 – XOR Operation in an Array



1486. XOR Operation in an Array

Easy 495 221 Add to List Share

Given an integer `n` and an integer `start`.

Define an array `nums` where `nums[i] = start + 2*i` (0-indexed) and `n == nums.length`.

Return the bitwise XOR of all elements of `nums`.

Example 1:

```
Input: n = 5, start = 0
Output: 8
Explanation: Array nums is equal to [0, 2, 4, 6, 8] where (0 ^ 2 ^
4 ^ 6 ^ 8) = 8.
Where "^" corresponds to bitwise XOR operator.
```

Example 2:

```
Input: n = 4, start = 3
Output: 8
Explanation: Array nums is equal to [3, 5, 7, 9] where (3 ^ 5 ^ 7 ^
9) = 8.
```

https://leetcode.com/problems/xor-operation-in-an-array/