

[◀ Back to Chapter](#)[< Previous](#) [Next >](#)

Week 1: April 1st–April 7th ▼

Problems appear at midnight, Pacific



## Single Number

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

### Note:

Your algorithm should have a linear runtime complexity. Could you implement it without using extra memory?

### Example 1:

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

### Example 2:

**Input:** [4,1,2,1,2]  
**Output:** 4



Single Number



Happy Number



Maximum Subarray



Week 2: April 8th–April 14th ▶

The first problem for this section will



Week 3: April 15th–April 21st ▶

The first problem for this section will

Python3 ▼



```
1 ▼ class Solution:
2 ▼     def singleNumber(self, nums:
      List[int]) -> int:
3         acc = 0
4 ▼
```

**Week 4: April 22nd–April 28th** ▶

The first problem for this section will

**Week 5: April 29th–April 30th** ▶

--- -- - - - - -

```
        for i in
range(len(nums)):
5         acc = acc ^ nums[i]
6         return acc
```