

# 136 只出现一次的数字

## 题目描述

给定一个非空整数数组，除了某个元素只出现一次以外，其余每个元素均出现两次。找出那个只出现了一次的元素。

说明：

你的算法应该具有线性时间复杂度。 你可以不使用额外空间来实现吗？

示例 1:

```
1 输入: [2,2,1]
2 输出: 1
```

示例 2:

```
1 输入: [4,1,2,1,2]
2 输出: 4
```

## 代码

```
1 class Solution:
2     def singleNumber(self, nums: List[int]) -> int:
3         '''
4         任何数与0异或都是本身
5         任何数与本身异或都是0
6         另外异或满足交换和结合律
7         '''
8         xor = 0
9         for num in nums:
10             xor ^= num
11         return xor
```

成功 [显示详情 >](#)

执行用时 : **52 ms**, 在Single Number的Python3提交中击败了**94.52%** 的用户

内存消耗 : **14.8 MB**, 在Single Number的Python3提交中击败了**49.16%** 的用户

进行下一个挑战:

只出现一次的数字 II

只出现一次的数字 III

缺失数字

寻找重复数

找不同

炫耀一下:    

提交时间	状态	执行用时	内存消耗	语言
几秒前	通过	52 ms	14.8 MB	python3
16 分钟前	超出时间限制	N/A	N/A	python3