



链滴

# [每日 LeetCode] 27. Remove Element

作者: [Hanseltu](#)

原文链接: <https://ld246.com/article/1553437743733>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Description:

Given an array `nums` and a value `val`, remove all instances of that value [ **in-place** ] ([https://en.wikipedia.org/wiki/In-place\\_algorithm](https://en.wikipedia.org/wiki/In-place_algorithm)) and return the new length.

Do not allocate extra space for another array, you must do this by **modifying the input array in-place** with  $O(1)$  extra memory.

The order of elements can be changed. It doesn't matter what you leave beyond the new length.

### Example 1:

Given `_nums_ = [3,2,2,3]`, `_val_ = 3`,

Your function should return `length = 2`, with the first two elements of `_nums_` being 2.

It doesn't matter what you leave beyond the returned length.

### Example 2:

Given `nums = [0,1,2,2,3,0,4,2]`, `val = 2`,

Your function should return `length = 5`, with the first five elements of `nums` containing 0,1,3,0,4.

Note that the order of those five elements can be arbitrary.

It doesn't matter what values are set beyond the returned length.

### Clarification:

Confused why the returned value is an integer but your answer is an array?

Note that the input array is passed in by **reference**, which means modification to the input array will be known to the caller as well.

Internally you can think of this:

```
// nums is passed in by reference. (i.e., without making a copy)
int len = removeElement(nums, val);
```

```
// any modification to nums in your function would be known by the caller.
// using the length returned by your function, it prints the first **len** elements.
for (int i = 0; i < len; i++) {
    print(nums[i]);
}
```

---

思路：本题要求删除数组中为特定数值的元素，并返回最后的数组大小。遍历数组，将不等于val的元素依次放到数组的前面，最后调整新数组的长度。

---

## C++代码

```
class Solution {
public:
    int removeElement(vector<int>& nums, int val) {
        int count = 0;
        for (int i=0; i<nums.size(); i++){
            if (nums[i] != val)
                nums[count++] = nums[i];
        }
        nums.resize(count);
        return count;
    }
};
```

---

运行时间: 4ms

运行内存: 8.4M