

剑指 Offer 53 - I. 在排序数组中查找数字 I

Sunday, June 12, 2022 3:14 PM

<https://leetcode.cn/problems/zai-pai-xu-shu-zu-zhong-cha-zhao-shu-zi-lcof/>

简单

统计一个数字在排序数组中出现的次数。

示例 1:

输入: nums = [5,7,7,8,8,10], target = 8

输出: 2

示例 2:

输入: nums = [5,7,7,8,8,10], target = 6

输出: 0

nums 是一个非递减数组

我的思路:

1. 二分找该target, 找不到返回0。
2. 找到了, 再前后找, 计算个数。

```
class Solution {
    public int search(int[] nums, int target) {
        // 判空
        if(nums.length==0||nums==null) return 0;
        // 初始计数器、左坐标右做表
        int count=0;
        int l=0;
        int r=nums.length-1;
        // l,r 闭区间初始化写法, 故while循环加上等号判断。
        while(l<=r){
            int mid=l+(r-l)/2;
            // 找到了target
            if(nums[mid]==target){
                // 因为不确定此时target的位置, 故需要前后找
                // 找的时候mid坐标会变化, 故设cur保存当前mid坐标。
                int cur=mid;
                // 先记上此时的一个
                count++;
                // 向左找
                mid--;
                while(mid>=0&&nums[mid]==target){
                    count++;
                    mid--;
                }
            }
        }
        return count;
    }
}
```

```

    }
    //向右找
    cur++;
    while(cur<nums.length&&nums[cur]==target){
        count++;
        cur++;
    }
    return count;
// 没找到 接着找
    }else if(nums[mid]<target){
        l=mid+1;
    }else{
        r=mid-1;
    }
}
return 0;
}
}

```

剑指 Offer 53 - II. 0 ~ n-1中缺失的数字

<https://leetcode.cn/problems/que-shi-de-shu-zi-lcof/>

简单

一个长度为n-1的递增排序数组中的所有数字都是唯一的，并且每个数字都在范围0 ~ n-1之内。在范围0 ~ n-1内的n个数字中有且只有一个数字不在该数组中，请找出这个数字。

示例 1:

输入: [0,1,3]

输出: 2

示例 2:

输入: [0,1,2,3,4,5,6,7,9]

输出: 8

解法一：HashMap，for循环遍历数组放入，再for循环遍历0-n查找。O(N)O(N)

解法二：先排序，再for循环。O(nlogn)O(1)

解法三：数组求和，0-n求和公式求和，二者之差即为该数。O(N)O(1)