

724 Find Pivot Index

2018年5月4日 12:35

Given an array of integers `nums`, write a method that returns the "pivot" index of this array.

We define the pivot index as the index where the sum of the numbers to the left of the index is equal to the sum of the numbers to the right of the index.

If no such index exists, we should return -1. If there are multiple pivot indexes, you should return the left-most pivot index.

Example 1:

Input:

`nums = [1, 7, 3, 6, 5, 6]`

Output: 3

Explanation:

The sum of the numbers to the left of index 3 (`nums[3] = 6`) is equal to the sum of numbers to the right of index 3.

Also, 3 is the first index where this occurs.

Example 2:

Input:

`nums = [1, 2, 3]`

Output: -1

Explanation:

There is no index that satisfies the conditions in the problem statement.

Note:

- The length of `nums` will be in the range `[0, 10000]`.
- Each element `nums[i]` will be an integer in the range `[-1000, 1000]`.

来自 <<https://leetcode.com/problems/find-pivot-index/description/>>

给定一个整数类型的数组 `nums`，请编写一个能够返回数组“中心索引”的方法。

我们是这样定义数组中心索引的：数组中心索引的左侧所有元素相加的和等于右侧所有元素相加的和。

如果数组不存在中心索引，那么我们应该返回 -1。如果数组有多个中心索引，那么我们应该返回最靠近左边的那一个。

说明:

- `nums` 的长度范围为 `[0, 10000]`。
- 任何一个 `nums[i]` 将会是一个范围在 `[-1000, 1000]` 的整数。

Solution for Python3:

```

1  class Solution:
2      def pivotIndex(self, nums):
3          """
4              :type nums: List[int]
5              :rtype: int
6              """
7          L, R = 0, sum(nums)
8          for k in range(len(nums)):
9              R -= nums[k]
10             if L == R:
11                 return k
12             L += nums[k]
13         return -1

```

Solution for C++:

```

1  class Solution {
2  public:
3      int pivotIndex(vector<int>& nums) {
4          int L = 0, R =
5          accumulate(nums.begin(), nums.end(), 0);
6          for (int k = 0; k < nums.size(); k++)
7          {
8              R -= nums[k];
9              if (L == R)
10                 return k;
11                 L += nums[k];
12             }
13         return -1;
14     }
15 };

```