

寻找数组的中心索引

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。

示例 1：

输入：

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

输出： 3

解释：

索引 3 (`nums[3] = 6`) 的左侧数之和 ($1 + 7 + 3 = 11$)，与右侧数之和 ($5 + 6 = 11$) 相等。

同时，3 也是第一个符合要求的中心索引。

示例 2：

输入：

`nums = [1, 2, 3]`

输出： -1

解释：

数组中不存在满足此条件的中心索引。

解题思路：

寻找中心索引，满足左半边和等于右半边和

同时整体满足 $sum = leftsum + rightsum + nums[i]$

根据上述这两个条件作为判断依据进行查找

```
int pivotIndex(int* nums, int numsSize){
    int leftsum=0;
    int rightsum=0;
    int sum=0;
    for(int i=0;i<numsSize;i++)
    {
```

```
        sum+=nums[i];
    }
    for(int i=0;i<numsSize;i++)
    {
        if(sum==2*leftsum+nums[i])
            return i;
        else
            leftsum+=nums[i];
    }
    return -1;
}
```