

462. 最少移动次数使数组元素相等 II

题目: Minimum Moves to Equal Array Elements II

语言: python3

英文版链接: <https://leetcode.com/problems/minimum-moves-to-equal-array-elements-ii/description/>

中文版链接: <https://leetcode-cn.com/problems/minimum-moves-to-equal-array-elements-ii/>

题目分析

每次可以对一个数组元素加一或者减一, 求最小的改变次数。

这是个典型的相遇问题, 移动距离最小的方式是所有元素都移动到中位数。理由如下:

设 m 为中位数。 a 和 b 是 m 两边的两个元素, 且 $b > a$ 。要使 a 和 b 相等, 它们总共移动的次数为 $b - a$, 这个值等于 $(b - m) + (m - a)$, 也就是把这两个数移动到中位数的移动次数。

设数组长度为 N , 则可以找到 $N/2$ 对 a 和 b 的组合, 使它们都移动到 m 的位置。

答案

根据上面的分析, 我们首先想到的就是排序然后找到中位数, 进而求解, 这样的时间复杂度是 $O(N\log N)$ 。

```
def minMoves2(self, nums) -> int:
    # 首先进行排序
    nums = sorted(nums)
    # 结果变量
    res = 0
    # 设定左右两个指针, 分别指向首尾
    left, right = 0, len(nums) - 1
    while left < right:
        res += nums[right] - nums[left]
        left += 1
        right -= 1
    return res
```