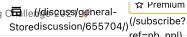
---(/) Explore Problems(/problemset/all/) Interview Contest Page 14 Problems(/problemset/all/) Problem







ref=nb_npl)





5832. Array With Elements Not Equal to Average of Neighbors

My Submissions (/contest/weekly-contest-254/problems/array-with-elements-not-equal-to-average-of-neighbors/submissions/)

Back to Contest (/contest/weekly-contest-254/)

You are given a 0-indexed array nums of distinct integers. You want to rearrange the elements in the array such that every element in the rearranged array is not equal to the average of its neighbors.

More formally, the rearranged array should have the property such that for every i in the range $1 \le i$ < nums.length -1, (nums[i-1] + nums[i+1]) / 2 is **not** equal to nums[i].

Return any rearrangement of nums that meets the requirements.

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Medium

Example 1:

```
Input: nums = [1,2,3,4,5]
Output: [1,2,4,5,3]
Explanation:
When i=1, nums[i] = 2, and the average of its neighbors is (1+4) / 2 = 2.5.
When i=2, nums[i] = 4, and the average of its neighbors is (2+5) / 2 = 3.5.
When i=3, nums[i] = 5, and the average of its neighbors is (4+3) / 2 = 3.5.
```

Example 2:

```
Input: nums = [6,2,0,9,7]
Output: [9,7,6,2,0]
Explanation:
When i=1, nums[i] = 7, and the average of its neighbors is (9+6) / 2 = 7.5.
When i=2, nums[i] = 6, and the average of its neighbors is (7+2) / 2 = 4.5.
When i=3, nums[i] = 2, and the average of its neighbors is (6+0) / 2 = 3.
```

Constraints:

- 3 <= nums.length <= 10⁵
- $0 \le nums[i] \le 10^5$

```
JavaScript
                                                                                                                             \boldsymbol{\varepsilon}
    const stin = (a) \Rightarrow a.sort((x, y) \Rightarrow x - y);
    const swap = (a, i, j) \Rightarrow [a[i], a[j]] = [a[j], a[i]];
 3 v const rearrangeArray = (a) ⇒ {
 4
         let n = a.length;
 5
          stin(a);
 6 ▼
          for (let i = 1; i + 1 < n; i += 2) {
 7
              swap(a, i, i + 1);
 8
 9
         // pr(test(a));
10
         return a;
11
    };
```