6006. Removing Minimum Number of Magic Beans

My Submissions (/contest/weekly-contest-280/problems/removing-minimum-number-of-magic-beans/submissions/)

Back to Contest (/contest/weekly-contest-280/)

You are given an array of **positive** integers `beans`, where each integer represents the number of magic beans found in a particular magic bag.

Remove any number of beans (**possibly none**) from each bag such that the number of beans in each remaining **non-empty** bag (still containing **at least one** bean) is **equal**. Once a bean has been removed from a bag, you are **not** allowed to return it to any of the bags.

Return *the **minimum** number of magic beans that you have to remove*.

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

Example 1:

```
Input: beans = [4,1,6,5]
Output: 4
Explanation:
- We remove 1 bean from the bag with only 1 bean.
  This results in the remaining bags: [4,0,6,5]
- Then we remove 2 beans from the bag with 6 beans.
  This results in the remaining bags: [4,0,4,5]
- Then we remove 1 bean from the bag with 5 beans.
  This results in the remaining bags: [4,0,4,4]
We removed a total of 1 + 2 + 1 = 4 beans to make the remaining non-empty bags have an equal number of beans.
There are no other solutions that remove 4 beans or fewer.
```

Example 2:

```
Input: beans = [2,10,3,2]
Output: 7
Explanation:
- We remove 2 beans from one of the bags with 2 beans.
  This results in the remaining bags: [0,10,3,2]
- Then we remove 2 beans from the other bag with 2 beans.
  This results in the remaining bags: [0,10,3,0]
- Then we remove 3 beans from the bag with 3 beans.
  This results in the remaining bags: [0,10,0,0]

We removed a total of 2 + 2 + 3 = 7 beans to make the remaining non-empty bags have an equal number of beans.
There are no other solutions that removes 7 beans or fewer.
```

Constraints:

- `1 <= beans.length <= 10^5`
- `1 <= beans[i] <= 10^5`

JavaScript

```
const preSum = (a) => { let pre = [0]; for (let i = 0; i < a.length; i++) { pre.push(pre[i] + a[i]); } return pre; };
const subArraySum = (a, l, r) => a[r + 1] - a[l];

const minimumRemoval = (a) => {
    a.sort((x, y) => x - y);
    let n = a.length, pre = preSum(a), res = Number.MAX_SAFE_INTEGER;
    for (let i = 0; i < n; i++) {
        let lsum = subArraySum(pre, 0, i -1), rsum = subArraySum(pre, i, n - 1), len = n - i;
        let move = lsum + (rsum - len * a[i]);
        // pr(a[i], a.slice(0, i), a.slice(i), lsum, rsum, len, move);
        res = Math.min(res, move);
    }
    return res;
```

```
14 };
```

Custom Testcase Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/640286306/) More Details (/submissions/detail/640286306/)

Share your acceptance!