1664. Ways to Make a Fair Array

Description

You are given an integer array `nums`. You can choose **exactly one** index (**0-indexed**) and remove the element. Notice that the index of the elements may change after the removal.

For example, if `nums = [6,1,7,4,1]`:

- Choosing to remove index `1` results in `nums = [6,7,4,1]`.
- Choosing to remove index `2` results in `nums = [6,1,4,1]`.
- Choosing to remove index `4` results in `nums = [6,1,7,4]`.

An array is **fair** if the sum of the odd-indexed values equals the sum of the even-indexed values.

Return the ***number*** *of indices that you could choose such that after the removal,* `nums` *is* ***fair****.*

Example 1:

```
Input: nums = [2,1,6,4]
Output: 1
Explanation:
Remove index 0: [1,6,4] -> Even sum: 1 + 4 = 5. Odd sum: 6. Not fair.
Remove index 1: [2,6,4] -> Even sum: 2 + 4 = 6. Odd sum: 6. Fair.
Remove index 2: [2,1,4] -> Even sum: 2 + 4 = 6. Odd sum: 1. Not fair.
Remove index 3: [2,1,6] -> Even sum: 2 + 6 = 8. Odd sum: 1. Not fair.
There is 1 index that you can remove to make nums fair.
```

Example 2:

```
Input: nums = [1,1,1]
Output: 3
Explanation: You can remove any index and the remaining array is fair.
```

Example 3:

```
Input: nums = [1,2,3]
Output: 0
Explanation: You cannot make a fair array after removing any index.
```

Constraints:

- `1 <= nums.length <= 10^5`
- `1 <= nums[i] <= 10^4`