Given an array of integers called nums, you can perform **any** of the following operation while nums contains **at least** 2 elements:

* Choose the first two elements of nums and delete them.
* Choose the last two elements of nums and delete them.
* Choose the first and the last elements of nums and delete them.

The **score** of the operation is the sum of the deleted elements.

Your task is to find the **maximum** number of operations that can be performed, such that **all operations have the same score**.

Return *the* ***maximum*** *number of operations possible that satisfy the condition mentioned above*.

**Example 1:**

Input: nums = [3,2,1,2,3,4]  
Output: 3  
Explanation: We perform the following operations:  
- Delete the first two elements, with score 3 + 2 = 5, nums = [1,2,3,4].  
- Delete the first and the last elements, with score 1 + 4 = 5, nums = [2,3].  
- Delete the first and the last elements, with score 2 + 3 = 5, nums = [].  
We are unable to perform any more operations as nums is empty.

**Example 2:**

Input: nums = [3,2,6,1,4]  
Output: 2  
Explanation: We perform the following operations:  
- Delete the first two elements, with score 3 + 2 = 5, nums = [6,1,4].  
- Delete the last two elements, with score 1 + 4 = 5, nums = [6].  
It can be proven that we can perform at most 2 operations.

**Constraints:**

* 2 <= nums.length <= 2000
* 1 <= nums[i] <= 1000