You are given a **0-indexed** integer array nums.

**Swaps** of **adjacent** elements are able to be performed on nums.

A **valid** array meets the following conditions:

* The largest element (any of the largest elements if there are multiple) is at the rightmost position in the array.
* The smallest element (any of the smallest elements if there are multiple) is at the leftmost position in the array.

Return *the* ***minimum*** *swaps required to make* nums *a valid array*.

**Example 1:**

Input: nums = [3,4,5,5,3,1]  
Output: 6  
Explanation: Perform the following swaps:  
- Swap 1: Swap the 3rd and 4th elements, nums is then [3,4,5,3,5,1].  
- Swap 2: Swap the 4th and 5th elements, nums is then [3,4,5,3,1,5].  
- Swap 3: Swap the 3rd and 4th elements, nums is then [3,4,5,1,3,5].  
- Swap 4: Swap the 2nd and 3rd elements, nums is then [3,4,1,5,3,5].  
- Swap 5: Swap the 1st and 2nd elements, nums is then [3,1,4,5,3,5].  
- Swap 6: Swap the 0th and 1st elements, nums is then [1,3,4,5,3,5].  
It can be shown that 6 swaps is the minimum swaps required to make a valid array.

**Example 2:**

Input: nums = [9]  
Output: 0  
Explanation: The array is already valid, so we return 0.

**Constraints:**

* 1 <= nums.length <= 105
* 1 <= nums[i] <= 105