The **score** of an array is defined as the **product** of its sum and its length.

* For example, the score of [1, 2, 3, 4, 5] is (1 + 2 + 3 + 4 + 5) \* 5 = 75.

Given a positive integer array nums and an integer k, return *the* ***number of non-empty subarrays*** *of* nums *whose score is* ***strictly less*** *than* k.

A **subarray** is a contiguous sequence of elements within an array.

**Example 1:**

Input: nums = [2,1,4,3,5], k = 10  
Output: 6  
Explanation:  
The 6 subarrays having scores less than 10 are:  
- [2] with score 2 \* 1 = 2.  
- [1] with score 1 \* 1 = 1.  
- [4] with score 4 \* 1 = 4.  
- [3] with score 3 \* 1 = 3.   
- [5] with score 5 \* 1 = 5.  
- [2,1] with score (2 + 1) \* 2 = 6.  
Note that subarrays such as [1,4] and [4,3,5] are not considered because their scores are 10 and 36 respectively, while we need scores strictly less than 10.

**Example 2:**

Input: nums = [1,1,1], k = 5  
Output: 5  
Explanation:  
Every subarray except [1,1,1] has a score less than 5.  
[1,1,1] has a score (1 + 1 + 1) \* 3 = 9, which is greater than 5.  
Thus, there are 5 subarrays having scores less than 5.

**Constraints:**

* 1 <= nums.length <= 105
* 1 <= nums[i] <= 105
* 1 <= k <= 1015