**[Subarray Product Less Than K](https://leetcode.com/problems/subarray-product-less-than-k/)**

Given an array of integers nums and an integer k, return *the number of contiguous subarrays where the product of all the elements in the subarray is strictly less than*k.

**Example 1:**

**Input:** nums = [10,5,2,6], k = 100

**Output:** 8

**Explanation:** The 8 subarrays that have product less than 100 are:

[10], [5], [2], [6], [10, 5], [5, 2], [2, 6], [5, 2, 6]

Note that [10, 5, 2] is not included as the product of 100 is not strictly less than k.

**Example 2:**

**Input:** nums = [1,2,3], k = 0

**Output:** 0

**Constraints:**

* 1 <= nums.length <= 3 \* 104
* 1 <= nums[i] <= 1000
* 0 <= k <= 106

class Solution {

public:

int numSubarrayProductLessThanK(vector<int>& nums, int k) {

int left = 0, right = 0, product = 1, count = 0;

int n = nums.size();

if(k <= 1) return 0;

while (right < n) {

product \*= nums[right];

while (product >= k) product /= nums[left++];

count += 1 + (right - left);

right++;

}

return count;

}

};

Link : <https://leetcode.com/problems/subarray-product-less-than-k/submissions/1215634405/?envType=daily-question&envId=2024-03-27>