[LeetCode](https://leetcode.com/problems/count-subarrays-with-fixed-bounds/description/?envType=daily-question&envId=2024-03-31)

<https://github.com/AdityaKonda6/-50DaysOfCoding>

<https://leetcode.com/problems/count-subarrays-with-fixed-bounds/description/?envType=daily-question&envId=2024-03-31>

<https://www.linkedin.com/in/aditya-adi-konda/>

Day 41 of [#50dayscodingchallenge](https://www.linkedin.com/feed/hashtag/?keywords=50dayscodingchallenge&highlightedUpdateUrns=urn:li:activity:7166316239483461633):  
[#leetcode](https://www.linkedin.com/feed/hashtag/?keywords=leetcode&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcodechallenge](https://www.linkedin.com/feed/hashtag/?keywords=leetcodechallenge&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcodestreak](https://www.linkedin.com/feed/hashtag/?keywords=leetcodestreak&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcode2024](https://www.linkedin.com/feed/hashtag/?keywords=leetcode2024&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcode50day](https://www.linkedin.com/feed/hashtag/?keywords=leetcode50day&highlightedUpdateUrns=urn:li:activity:7166316239483461633)  
   
Ventured further into my coding journey today, tackling the engaging LeetCode Problem "Successfully solved LeetCode Problem ���"

“2444. Count Subarrays With Fixed Bounds.”

   
✨ Task: You are given an integer array nums and two integers minK and maxK.

A fixed-bound subarray of nums is a subarray that satisfies the following conditions:

The minimum value in the subarray is equal to minK.

The maximum value in the subarray is equal to maxK.

Return the number of fixed-bound subarrays.

A subarray is a contiguous part of an array.

Examples:

Example 1:

Input: nums = [1,3,5,2,7,5], minK = 1, maxK = 5

Output: 2

Explanation: The fixed-bound subarrays are [1,3,5] and [1,3,5,2].

Example 2:

Input: nums = [1,1,1,1], minK = 1, maxK = 1

Output: 10

Explanation: Every subarray of nums is a fixed-bound subarray. There are 10 possible subarrays.

Let's Connect:

If you find this problem intriguing or have insights to share, let's connect! I'm passionate about problem-solving, algorithmic thinking, and collaborative learning. Feel free to comment or reach out for engaging discussions and knowledge exchange.Unravel the mystery using your coding skills!

[#CodingChallenge](https://www.linkedin.com/feed/hashtag/?keywords=codingchallenge&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#Algorithm](https://www.linkedin.com/feed/hashtag/?keywords=algorithm&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#LinkedInPost](https://www.linkedin.com/feed/hashtag/?keywords=linkedinpost&highlightedUpdateUrns=urn:li:activity:7166316239483461633) #Algorithm #Optimization #DataStructures #CodingChallenge  
  
Excited about the progress and challenges ahead!  
   
Make Sure You Follow My GitHub For Solutions: <https://github.com/AdityaKonda6/-50DaysOfCoding>  
  
  
Happy coding!

**Solution:-**

class Solution {

  public long countSubarrays(int[] nums, int minK, int maxK) {

    long ans = 0;

    int j = -1;

    int prevMinKIndex = -1;

    int prevMaxKIndex = -1;

    for (int i = 0; i < nums.length; ++i) {

      if (nums[i] < minK || nums[i] > maxK)

        j = i;

      if (nums[i] == minK)

        prevMinKIndex = i;

      if (nums[i] == maxK)

        prevMaxKIndex = i;

      ans += Math.max(0, Math.min(prevMinKIndex, prevMaxKIndex) - j);

    }

    return ans;

  }

}



