**581. Shortest Unsorted Continuous Subarray**

Easy

150370FavoriteShare

Given an integer array, you need to find one **continuous subarray** that if you only sort this subarray in ascending order, then the whole array will be sorted in ascending order, too.

You need to find the **shortest** such subarray and output its length.

**Example 1:**

**Input:** [2, 6, 4, 8, 10, 9, 15]

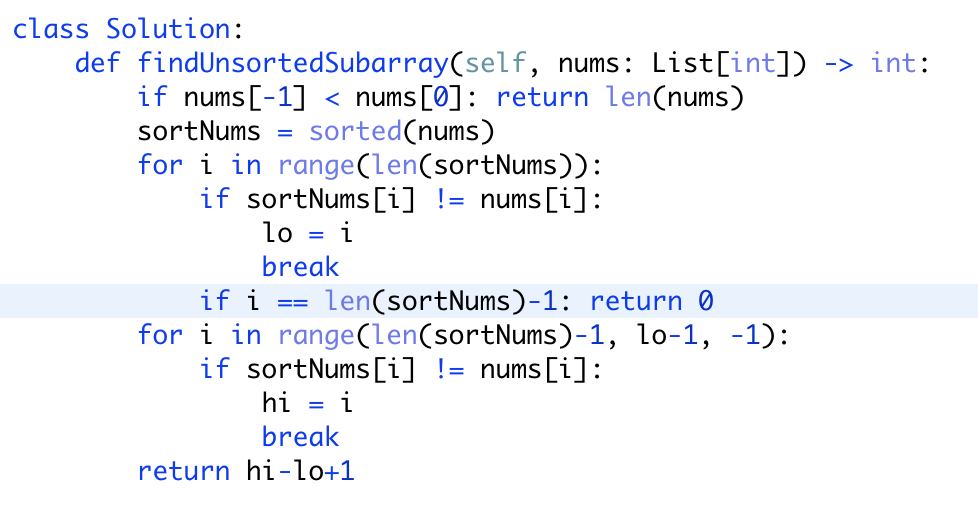
**Output:** 5

**Explanation:** You need to sort [6, 4, 8, 10, 9] in ascending order to make the whole array sorted in ascending order.

**Note:**

1. Then length of the input array is in range [1, 10,000].
2. The input array may contain duplicates, so ascending order here means **<=**.
3. Using sorted array

Comparing with sorted array and using two for loop to find the first and the last different number’s index. Return the difference plus one



* Time complexity : O(n\log n)*O*(*n*log*n*). Sorting takes n\log n*n*log*n* time.
* Space complexity : O(n)*O*(*n*). We are making copy of original array.