AlgoExpert Quad Layout Swift 12px Sublime Monokai 00:00:00

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1

```swift
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

class Program {
    // O(n) time | O(1) space
    func subarraySort(array: [Int]) -> [Int] {
        var minimumOutOfOrder = Int(Int16.max)
        var maximumOutOfOrder = -Int(Int16.max)

        for i in 0 ..< array.count {
            let currentNumber = array[i]

            if isOutOfOrder(i, array, currentNumber) {
                minimumOutOfOrder = min(currentNumber, minimumOutOfOrder)
                maximumOutOfOrder = max(currentNumber, maximumOutOfOrder)
            }
        }

        if minimumOutOfOrder == Int(Int16.max) {
            return [-1, -1]
        }

        var subarrayLeftIndex = 0
        while minimumOutOfOrder >= array[subarrayLeftIndex] {
            subarrayLeftIndex += 1
        }

        var subarrayRightIndex = array.count - 1
        while maximumOutOfOrder <= array[subarrayRightIndex] {
            subarrayRightIndex -= 1
        }

        return [subarrayLeftIndex, subarrayRightIndex]
    }

    func isOutOfOrder(_ i: Int, _ array: [Int], _ currentNumber: Int) -> Bool {
        if i == 0 {
            return currentNumber > array[i + 1]
        } else if i == array.count - 1 {
            return currentNumber < array[i - 1]
        } else {
            return currentNumber > array[i + 1] || currentNumber < array[i - 1]
        }
    }
}
```