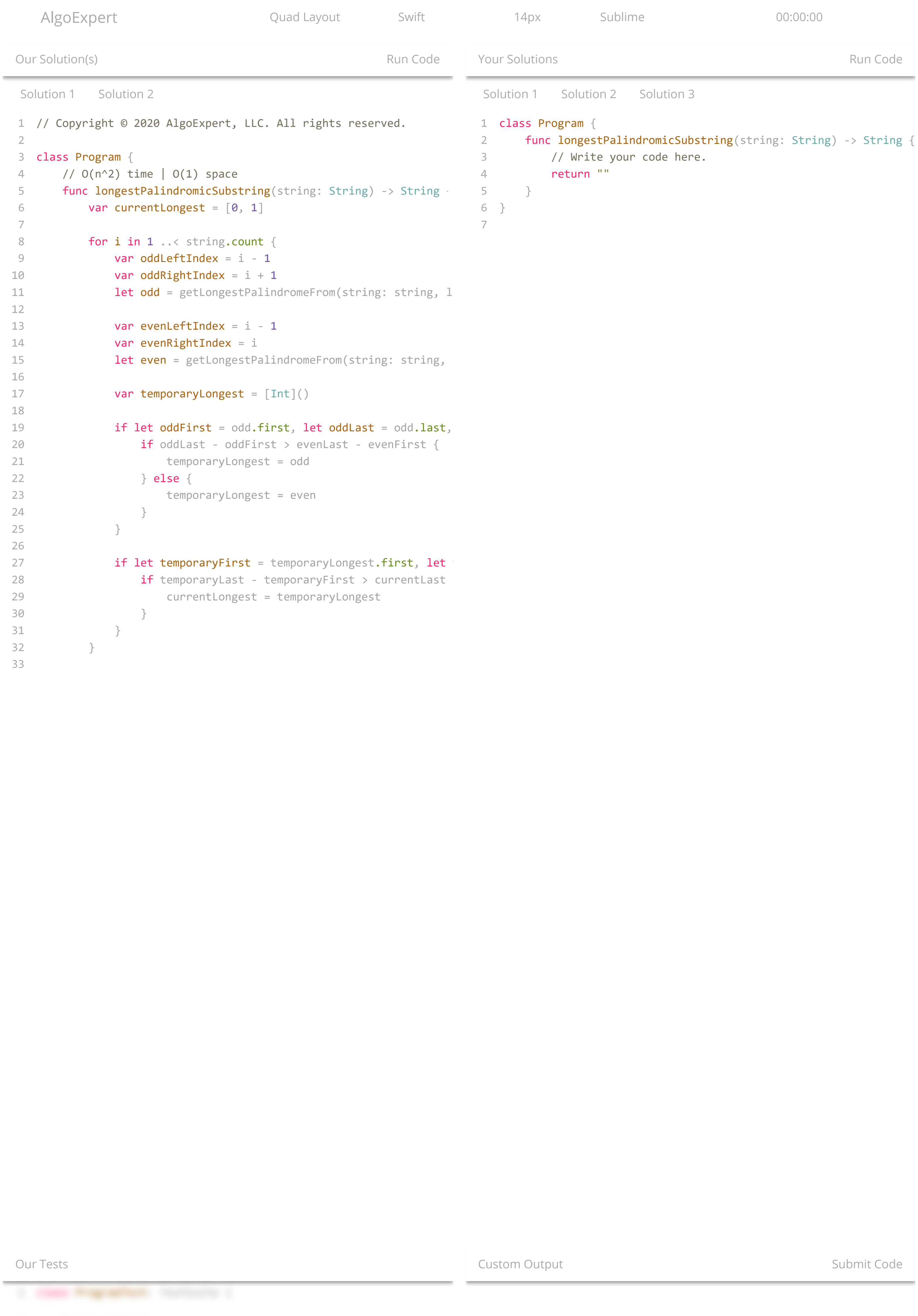

```swift
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

class Program {
    // O(n^2) time | O(1) space
    func longestPalindromicSubstring(string: String) -> String
        var currentLongest = [0, 1]

        for i in 1 ..< string.count {
            var oddLeftIndex = i - 1
            var oddRightIndex = i + 1
            let odd = getLongestPalindromeFrom(string: string, l

            var evenLeftIndex = i - 1
            var evenRightIndex = i
            let even = getLongestPalindromeFrom(string: string,

            var temporaryLongest = [Int]()

            if let oddFirst = odd.first, let oddLast = odd.last,
                if oddLast - oddFirst > evenLast - evenFirst {
                    temporaryLongest = odd
                } else {
                    temporaryLongest = even
                }
            }

            if let temporaryFirst = temporaryLongest.first, let
                if temporaryLast - temporaryFirst > currentLast
                    currentLongest = temporaryLongest
                }
            }
        }

```

```swift
class Program {
    func longestPalindromicSubstring(string: String) -> String {
        // Write your code here.
        return ""
    }
}

```

Run or submit code when you're ready.