Prompt | Scratchpad | Our Solution(s) | Video Explanation | Run Code

Solution 1 | Solution 2

```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, leftIndex: &oddLeftIndex, ri

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

            var temporaryLongest = [Int]()

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

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

        let firstIndex = string.index(string.startIndex, offsetBy: currentLongest.first!)
        let lastIndex = string.index(string.startIndex, offsetBy: currentLongest.last!)
        let result = String(string[firstIndex ..< lastIndex])

        return result
    }

    func getLongestPalindromeFrom(string: String, leftIndex: inout Int, rightIndex: inout I
        while leftIndex >= 0, rightIndex < string.count {
            let leftStringIndex = string.index(string.startIndex, offsetBy: leftIndex)
            let rightStringIndex = string.index(string.startIndex, offsetBy: rightIndex)

            if string[leftStringIndex] != string[rightStringIndex] {
                break
            }

            leftIndex -= 1
            rightIndex += 1
        }

        return [leftIndex + 1, rightIndex]
    }
}

```

Your Solutions | Run Code

Solution 1 | Solution 2 | Solution 3

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

```

Custom Output | Raw Output | Submit Code

Run or submit code when you're ready.