Our Solution(s) Run Code

Solution 1

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

class Program {
    class Chain {
        var nextString: String
        var maxChainLength: Int

        init(_ nextString: String, _ maxChainLength: Int) {
            self.nextString = nextString
            self.maxChainLength = maxChainLength
        }
    }

    // O(n * m^2 + nlog(n)) time | O(nm) space - where n is the number
    // m is the length of the longest string
    func longestStringChain(_ strings: [String]) -> [String] {
        // For every string, imagine the longest string chain that sta
        // Set up every string to point to the next string in its resp
        // string chain. Also keep track of the lengths of these longe
        var stringChains = [String: Chain]()
        for str in strings {
            stringChains[str] = Chain("", 1)
        }

        // Sort the strings based on their length so that whenever we
        // string (as we iterate through them from left to right), we
        // already have computed the longest string chains of any smal
        let sortedStrings = strings.sorted {
            $0.length < $1.length
        }

        for str in sortedStrings {
            findLongestStringChain(str, &stringChains)
```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```swift
class Program {
    func longestStringChain(_ strings: [String]) -> [String] {
        // Write your code here.
        return []
    }
}

```

Our Tests

Custom Output Submit Code

Run or submit code when you're ready.