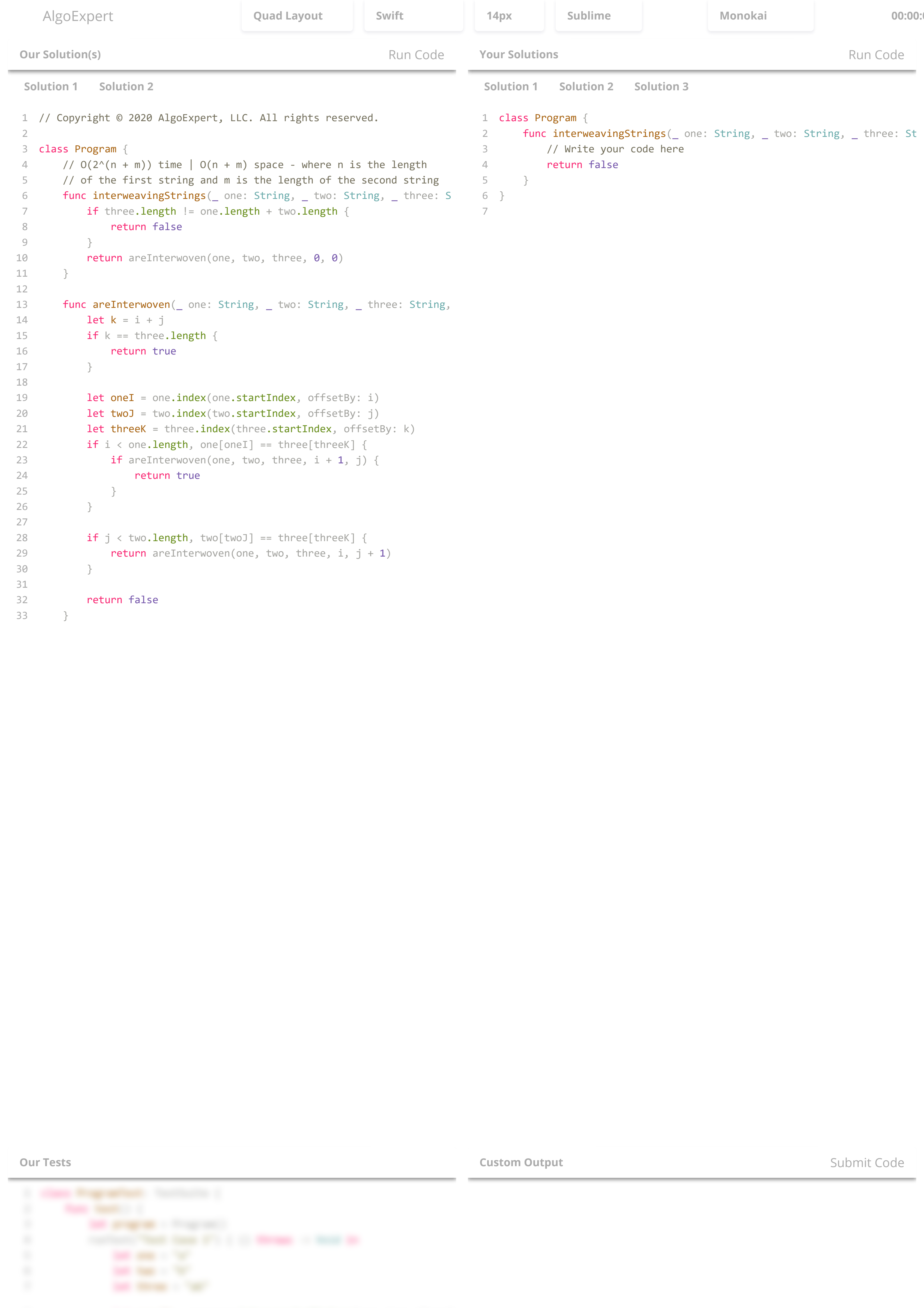

```swift
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

class Program {
    // O(2^(n + m)) time | O(n + m) space - where n is the length
    // of the first string and m is the length of the second string
    func interweavingStrings(_ one: String, _ two: String, _ three: S
        if three.length != one.length + two.length {
            return false
        }
        return areInterwoven(one, two, three, 0, 0)
    }

    func areInterwoven(_ one: String, _ two: String, _ three: String,
        let k = i + j
        if k == three.length {
            return true
        }

        let oneI = one.index(one.startIndex, offsetBy: i)
        let twoJ = two.index(two.startIndex, offsetBy: j)
        let threeK = three.index(three.startIndex, offsetBy: k)
        if i < one.length, one[oneI] == three[threeK] {
            if areInterwoven(one, two, three, i + 1, j) {
                return true
            }
        }

        if j < two.length, two[twoJ] == three[threeK] {
            return areInterwoven(one, two, three, i, j + 1)
        }

        return false
    }
```

```swift
class Program {
    func interweavingStrings(_ one: String, _ two: String, _ three: St
        // Write your code here
        return false
    }
}
```

Run or submit code when you're ready.