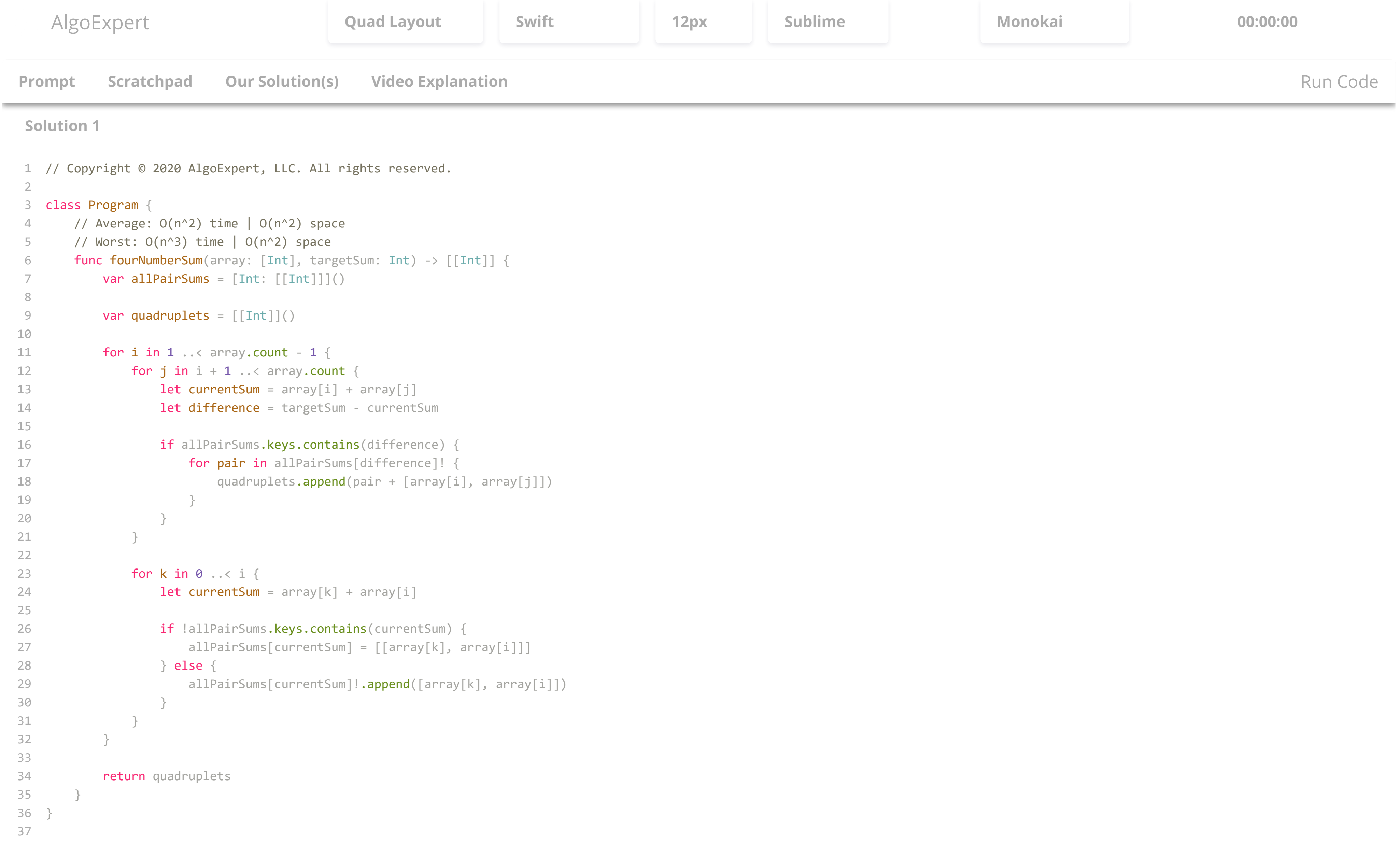

Solution 1

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

class Program {
    // Average: O(n^2) time | O(n^2) space
    // Worst: O(n^3) time | O(n^2) space
    func fourNumberSum(array: [Int], targetSum: Int) -> [[Int]] {
        var allPairSums = [Int: [[Int]]]()

        var quadruplets = [[Int]]()

        for i in 1 ..< array.count - 1 {
            for j in i + 1 ..< array.count {
                let currentSum = array[i] + array[j]
                let difference = targetSum - currentSum

                if allPairSums.keys.contains(difference) {
                    for pair in allPairSums[difference]! {
                        quadruplets.append(pair + [array[i], array[j]])
                    }
                }
            }

            for k in 0 ..< i {
                let currentSum = array[k] + array[i]

                if !allPairSums.keys.contains(currentSum) {
                    allPairSums[currentSum] = [[array[k], array[i]]]
                } else {
                    allPairSums[currentSum]!.append([array[k], array[i]])
                }
            }
        }

        return quadruplets
    }
}
```