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

Solution 1 | Solution 2

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

class Program {
    // O(Log(n)) time | O(1) space
    func binarySearch(array: [Int], target: Int) -> Int {
        var leftPointer = 0
        var rightPointer = array.count - 1
        return binarySearchHelper(array: array, target: target, leftPointer: &lef
    }

    func binarySearchHelper(array: [Int], target: Int, leftPointer: inout Int, ri
        while leftPointer <= rightPointer {
            let middle = (leftPointer + rightPointer) / 2
            let potentialMatch = array[middle]
            if target == potentialMatch {
                return middle
            } else if target < potentialMatch {
                rightPointer = middle - 1
            } else {
                leftPointer = middle + 1
            }
        }

        return -1
    }
}

```

Your Solutions | Run Code

Solution 1 | Solution 2 | Solution 3

```swift
class Program {
    func binarySearch(array: [Int], target: Int) -> Int {
        // Write your code here.
        return -1
    }
}

```

Custom Output | Raw Output | Submit Code

Run or submit code when you're ready.