

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

Solution 1	Solution 2	Solution 3
<pre>1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved. 2 3 ▾ class Program { 4 // Average case: when the created BST is balanced 5 // O(nlog(n)) time O(n) space - where n is the length of the array 6 // --- 7 // Worst case: when the the created BST is like a linked list 8 // O(n^2) time O(n) space 9 ▾ static func rightSmallerThan(_ array: [Int]) -> [Int] { 10 ▾ if array.count == 0 { 11 return [Int]() 12 } 13 14 let lastIdx = array.count - 1 15 var bst = SpecialBST(array[lastIdx], lastIdx, 0) 16 ▾ for i in (0 ..< lastIdx).reversed() { 17 bst.insert(array[i], i) 18 } 19 20 var rightSmallerCounts = [Int]() 21 ▾ for i in array { 22 rightSmallerCounts.append(i) 23 } 24 getRightSmallerCounts(bst, &rightSmallerCounts) 25 return rightSmallerCounts 26 } 27 28 ▾ static func getRightSmallerCounts(_ bst: SpecialBST?, _ rightSmallerCounts: inout [Int]) { 29 ▾ if let tree = bst { 30 rightSmallerCounts[tree.idx] = tree.numSmallerAtInsertTime 31 getRightSmallerCounts(tree.left, &rightSmallerCounts) 32 getRightSmallerCounts(tree.right, &rightSmallerCounts) 33 } 34 } 35 36 ▾ class SpecialBST { 37 var value: Int 38 var idx: Int 39 var numSmallerAtInsertTime: Int 40 41 var leftSubtreeSize: Int = 0 42 var left: SpecialBST? 43 var right: SpecialBST? 44 45 ▾ init(_ value: Int, _ idx: Int, _ numSmallerAtInsertTime: Int) { 46 self.value = value 47 self.idx = idx 48 self.numSmallerAtInsertTime = numSmallerAtInsertTime 49 } 50 51 ▾ func insert(_ value: Int, _ idx: Int, _ numSmallerAtInsertTime: Int = 0) { 52 ▾ if value < self.value { 53 leftSubtreeSize += 1 54 ▾ if let left = self.left { 55 left.insert(value, idx, numSmallerAtInsertTime) 56 ▾ } else { 57 left = SpecialBST(value, idx, numSmallerAtInsertTime) 58 } 59 return 60 } 61 62 var numSmaller = numSmallerAtInsertTime + leftSubtreeSize 63 ▾ if value > self.value { 64 numSmaller += 1 65 } 66 67 ▾ if let right = self.right { 68 right.insert(value, idx, numSmaller) 69 ▾ } else { 70 right = SpecialBST(value, idx, numSmaller) 71 } 72 } 73 } 74 } 75 }</pre>		

