

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 package main 4 5 // O(n) time   O(n) space - where n is the length of the array 6 ▾ func MinHeightBST(array []int) *BST { 7     return constructMinHeightBst(array, 0, len(array)-1) 8 } 9 10 ▾ func constructMinHeightBst(array []int, startIdx, endIdx int) *BST { 11 ▾ if endIdx &lt; startIdx { 12     return nil 13 } 14 midIdx := (startIdx + endIdx) / 2 15 bst := &amp;BST{Value: array[midIdx]} 16 bst.Left = constructMinHeightBst(array, startIdx, midIdx-1) 17 bst.Right = constructMinHeightBst(array, midIdx+1, endIdx) 18 return bst 19 } 20 21 ▾ type BST struct { 22     Value int 23 24     Left *BST 25     Right *BST 26 } 27 28 // We don't use this method for this solution. 29 ▾ func (tree *BST) Insert(value int) *BST { 30 ▾ if value &lt; tree.Value { 31 ▾ if tree.Left == nil { 32     tree.Left = &amp;BST{Value: value} 33 ▾ } else { 34     tree.Left.Insert(value) 35 } 36 ▾ } else { 37 ▾ if tree.Right == nil { 38     tree.Right = &amp;BST{Value: value} 39 ▾ } else { 40     tree.Right.Insert(value) 41 } 42 } 43 return tree 44 } 45</pre>		

