AlgoExpert Quad Layout Go 12px Sublime Monokai 00:00:00

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1 Solution 2

```
_{\rm 1} \, // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 package main
 5 // Best: O(nlog(n)) time | O(nlog(n)) space
 6 // Average: O(nlog(n)) time | O(nlog(n)) space
 7 // Worst: O(nlog(n)) time | O(nlog(n)) space
8 func MergeSort(array []int) []int {
9 if len(array) <= 1 {
10
          return array
11
12
       middleIndex := len(array) / 2
13
       leftHalf := MergeSort(array[:middleIndex])
       rightHalf := MergeSort(array[middleIndex:])
14
15
       return mergeSortedArrays(leftHalf, rightHalf)
16 }
17
18
      func mergeSortedArrays(leftHalf, rightHalf []int) []int {
19
       sortedArray := make([]int, len(leftHalf)+len(rightHalf))
       k, i, j := 0, 0, 0
for i < len(leftHalf) && j < len(rightHalf) {
   if leftHalf[i] <= rightHalf[j] {</pre>
20
21
22
23
            sortedArray[k] = leftHalf[i]
24
25
          } else {
26
            sortedArray[k] = rightHalf[j]
27
28
29
            j++
          k++
30
31
        for i < len(leftHalf) {</pre>
32
         sortedArray[k] = leftHalf[i]
33
          i++
34
          k++
35
36
        \quad \textbf{for} \ \texttt{j} \ \texttt{<} \ \texttt{len}(\texttt{rightHalf}) \ \texttt{\{}
          sortedArray[k] = rightHalf[j]
37
38
39
40
41
        return sortedArray
42 }
43
```