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

 Prompt
 Scratchpad
 Our Solution(s)
 Video Explanation

Run Code

```
1\, // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
     class Program {
         // Best: O(nlog(n)) time | O(nlog(n)) space
         // Average: O(n\log(n)) time | O(n\log(n)) space
         // Worst: O(nlog(n)) time | O(nlog(n)) space
         func mergeSort(_ array: inout [Int]) -> [Int] {
   if array.count <= 1 {</pre>
                  return array
10
11
12
             let middleIndex = Int(Double(array.count / 2).rounded(.down))
13
             var leftHalf = Array(array[0 ..< middleIndex])</pre>
14
              var rightHalf = Array(array[middleIndex ..< array.count])</pre>
15
16
              {\bf return} \ {\tt mergeSortedArrays(mergeSort(\&leftHalf), mergeSort(\&rightHalf))}
17
18
         func mergeSortedArrays(_ leftHalf: [Int], _ rightHalf: [Int]) -> [Int] {
    var sortedArray = Array(repeating: 0, count: leftHalf.count + rightHalf.count)
19
20
21
22
              var k = 0, i = 0, j = 0
23
24
              while i < leftHalf.count, j < rightHalf.count {</pre>
25
                  if leftHalf[i] <= rightHalf[j] {</pre>
26
                       sortedArray[k] = leftHalf[i]
27
                       i += 1
28
                  } else {
29
                       sortedArray[k] = rightHalf[j]
30
                       j += 1
31
32
33
                  k += 1
34
35
              while i < leftHalf.count {</pre>
36
37
                  sortedArray[k] = leftHalf[i]
38
                  i += 1
39
                  k += 1
40
41
              while j < rightHalf.count {</pre>
42
                  sortedArray[k] = rightHalf[j]
43
44
                  j += 1
45
                  k += 1
46
47
              return sortedArray
48
49
50 }
```

Solution 1 Solution 2