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

 Prompt
 Scratchpad
 Our Solution(s)
 Video Explanation

Run Code

```
Solution 1 Solution 2
```

43

```
{\tt 1} {\tt \#} Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 4 # Average: O(nlog(n)) time | O(n) space
 5 # Worst: O(nlog(n)) time | O(n) space
 6 def mergeSort(array):
        if len(array) <= 1:</pre>
           return array
         auxiliaryArray = array[:]
        \verb|mergeSortHelper(array, 0, len(array) - 1, auxiliaryArray)|\\
10
11
12
13
14 def mergeSortHelper(mainArray, startIdx, endIdx, auxiliaryArray):
15
        if startIdx == endIdx:
16
           return
17
         middleIdx = (startIdx + endIdx) // 2
        mergeSortHelper(auxiliaryArray, startIdx, middleIdx, mainArray)
18
19
        mergeSortHelper(auxiliaryArray, middleIdx + 1, endIdx, mainArray)
20
21
        doMerge(mainArray, startIdx, middleIdx, endIdx, auxiliaryArray)
22
23 def doMerge(mainArray, startIdx, middleIdx, endIdx, auxiliaryArray):
24
25
        i = startIdx
26
        j = middleIdx + 1
27
28
        \label{eq:while} \mbox{ while i <= middleIdx and j <= endIdx:}
            if auxiliaryArray[i] <= auxiliaryArray[j]:</pre>
29
                mainArray[k] = auxiliaryArray[i]
30
                i += 1
31
            else:
32
              mainArray[k] = auxiliaryArray[j]
33
                j += 1
34
            k += 1
35
        while i <= middleIdx:</pre>
36
            mainArray[k] = auxiliaryArray[i]
37
            i += 1
38
            k += 1
39
         while j <= endIdx:</pre>
40
            mainArray[k] = auxiliaryArray[j]
41
            j += 1
42
            k += 1
```