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

Our Solution(s) Run Code Scratchpad Video Explanation

Solution 1 Solution 2

Prompt

```
1\, // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 using System.Linq;
 5 public class Program {
     // Best: O(nlog(n)) time | O(nlog(n)) space

// Average: O(nlog(n)) time | O(nlog(n)) space

// Worst: O(nlog(n)) time | O(nlog(n)) space
       public static int[] MergeSort(int[] array) {
         if (array.Length <= 1) {</pre>
10
11
           return array;
12
         int middleIdx = array.Length / 2;
13
14
         int[] leftHalf = array.Take(middleIdx).ToArray();
15
         int[] rightHalf = array.Skip(middleIdx).ToArray();
         return mergeSortedArrays(MergeSort(leftHalf), MergeSort(rightHalf));
16
17
18
       public static int[] mergeSortedArrays(int[] leftHalf, int[] rightHalf) {
19
20
21
         int[] sortedArray = new int[leftHalf.Length + rightHalf.Length];
         int k = 0;
22
         int i = 0;
23
         int j = 0;
24
         while (i < leftHalf.Length && j < rightHalf.Length) {</pre>
25
           if (leftHalf[i] <= rightHalf[j]) {</pre>
26
             sortedArray[k++] = leftHalf[i++];
27
28
29
           } else {
             sortedArray[k++] = rightHalf[j++];
30
31
         while (i < leftHalf.Length) {</pre>
32
           sortedArray[k++] = leftHalf[i++];
33
34
35
         while (j < rightHalf.Length) {</pre>
           sortedArray[k++] = rightHalf[j++];
36
         return sortedArray;
38
39 }
40
```