

Our Solution(s)

Run Code

Your Solutions

Run Code

| Solution 1  | Solution 2   |            |
|---|--|------------|
| <pre>1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved. 2 3 class Program { 4     // Best: O(nlog(n) time   O(n) space 5     // Average: O(nlog(n) time   O(n) space 6     // Worst: O(nlog(n) time   O(n) space 7     func mergeSort(_ array: inout [Int]) -&gt; [Int] { 8         if array.count &lt;= 1 { 9             return array 10        } 11 12        var auxiliaryArray = array 13        mergeSortHelper(0, array.count - 1, &amp;array, &amp;auxiliaryArray) 14 15        return array 16    } 17 18    func mergeSortHelper(_ startIndex: Int, _ endIndex: Int, _ firstAr 19        if startIndex == endIndex { 20            return 21        } 22 23        let middleIndex = Int(Double((startIndex + endIndex) / 2).roun 24 25        mergeSortHelper(startIndex, middleIndex, &amp;secondArray, &amp;firstA 26        mergeSortHelper(middleIndex + 1, endIndex, &amp;secondArray, &amp;firs 27        doMerge(startIndex, middleIndex, endIndex, &amp;firstArray, &amp;secon 28    } 29 30    func doMerge(_ startIndex: Int, _ middleIndex: Int, _ endIndex: In 31        var k = startIndex, i = startIndex, j = middleIndex + 1 32 33        while i &lt;= middleIndex, j &lt;= endIndex {</pre> | <pre>1 class Program { 2     func mergeSort(_ array: inout [Int]) -&gt; [Int] { 3         // Write your code here. 4         return [] 5     } 6 } 7</pre> |            |
| Solution 1  | Solution 2   | Solution 3 |

Our Tests

Custom Output

Submit Code

```
1 class Program {
2     func mergeSort(_ array: inout [Int]) -> [Int] {
3         // Write your code here.
4         return []
5     }
6 }
7
```

