AlgoExpert Quad Layout C++ 12px Sublime Monokai 00:00:00

Run Code

Solution 1 Solution 2

Scratchpad

Prompt

41

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
 3 #include <vector>
 4 using namespace std;
 6 vector<int> mergeSortedArrays(vector<int> leftHalf, vector<int> rightHalf);
 8 // Best: O(nlog(n)) time | O(nlog(n)) space
 9 // Average: O(nlog(n)) time | O(nlog(n)) space
10 // Worst: O(nlog(n)) time | O(nlog(n)) space
11 vector<int> mergeSort(vector<int> array) {
      if (array.size() <= 1) {</pre>
12
13
        return array;
14
15
      int middleIdx = array.size() / 2;
16
      vector<int> leftHalf(array.begin(), array.begin() + middleIdx);
17
      vector<int> rightHalf(array.begin() + middleIdx, array.end());
18
      return mergeSortedArrays(mergeSort(leftHalf), mergeSort(rightHalf));
19 }
20
21 vector<int> mergeSortedArrays(vector<int> leftHalf, vector<int> rightHalf) {
22
      vector<int> sortedArray(leftHalf.size() + rightHalf.size(), 0);
23
      int k = 0;
24
      int i = 0;
      int j = 0;
25
26
      while (i < leftHalf.size() && j < rightHalf.size()) {</pre>
       if (leftHalf[i] <= rightHalf[j]) {
   sortedArray[k++] = leftHalf[i++];</pre>
27
28
29
        } else {
30
          sortedArray[k++] = rightHalf[j++];
31
32
33
      while (i < leftHalf.size()) {</pre>
34
35
        sortedArray[k++] = leftHalf[i++];
      while (j < rightHalf.size()) {</pre>
36
37
        sortedArray[k++] = rightHalf[j++];
38
39
      return sortedArray;
40 }
```

Our Solution(s)

Video Explanation