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

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1

49

```
1\, // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 class Program {
    // Best: O(nlog(n)) time | O(1) space
     // Average: O(nlog(n)) time | O(1) space
     // Worst: O(nlog(n)) time | O(1) space
      public static int[] heapSort(int[] array) {
       buildMaxHeap(array);
        for (int endIdx = array.length - 1; endIdx > 0; endIdx--) {
         swap(0, endIdx, array);
10
11
          siftDown(0, endIdx - 1, array);
12
13
       return array;
14
15
      public static void buildMaxHeap(int[] array) {
16
17
        int firstParentIdx = (array.length - 2) / 2;
18
        for (int currentIdx = firstParentIdx; currentIdx >= 0; currentIdx--) {
         siftDown(currentIdx, array.length - 1, array);
19
20
21
22
23
      public static void siftDown(int currentIdx, int endIdx, int[] heap) {
        int childOneIdx = currentIdx * 2 + 1;
24
25
        while (childOneIdx <= endIdx) {</pre>
26
         int childTwoIdx = currentIdx * 2 + 2 <= endIdx ? currentIdx * 2 + 2 : -1;</pre>
          int idxToSwap;
27
          if (childTwoIdx != -1 && heap[childTwoIdx] > heap[childOneIdx]) {
28
29
           idxToSwap = childTwoIdx;
30
         } else {
31
            idxToSwap = childOneIdx;
32
          if (heap[idxToSwap] > heap[currentIdx]) {
33
34
            swap(currentIdx, idxToSwap, heap);
35
            currentIdx = idxToSwap;
            childOneIdx = currentIdx * 2 + 1;
36
37
          } else {
38
           return;
39
40
41
42
43
      public static void swap(int i, int j, int[] array) {
        int temp = array[j];
44
        array[j] = array[i];
45
46
       array[i] = temp;
47
48 }
```