

PromptScratchpadOur Solution(s)Video Explanation

Run Code

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 // Best: O(nlog(n)) time | O(log(n)) space
4 // Average: O(nlog(n)) time | O(log(n)) space
5 // Worst: O(n^2) time | O(log(n)) space
6 function quickSort(array) {
7   quickSortHelper(array, 0, array.length - 1);
8   return array;
9 }
10
11 function quickSortHelper(array, startIdx, endIdx) {
12   if (startIdx >= endIdx) return;
13   const pivotIdx = startIdx;
14   let leftIdx = startIdx + 1;
15   let rightIdx = endIdx;
16   while (rightIdx >= leftIdx) {
17     if (array[leftIdx] > array[pivotIdx] && array[rightIdx] < array[pivotIdx]) {
18       swap(leftIdx, rightIdx, array);
19     }
20     if (array[leftIdx] <= array[pivotIdx]) leftIdx++;
21     if (array[rightIdx] >= array[pivotIdx]) rightIdx--;
22   }
23   swap(pivotIdx, rightIdx, array);
24   const leftSubarrayIsSmaller = rightIdx - 1 - startIdx < endIdx - (rightIdx + 1);
25   if (leftSubarrayIsSmaller) {
26     quickSortHelper(array, startIdx, rightIdx - 1);
27     quickSortHelper(array, rightIdx + 1, endIdx);
28   } else {
29     quickSortHelper(array, rightIdx + 1, endIdx);
30     quickSortHelper(array, startIdx, rightIdx - 1);
31   }
32 }
33
34 function swap(i, j, array) {
35   let temp = array[j];
36   array[j] = array[i];
37   array[i] = temp;
38 }
39
40 exports.quickSort = quickSort;
41
```