AlgoExpert Quad Layout JavaScript 12px Sublime Monok

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1

```javascript
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

// O(n) time | O(1) space - where n is the number of nodes in the Linked List
function rearrangeLinkedList(head, k) {
  let smallerListHead = null;
  let smallerListTail = null;
  let equalListHead = null;
  let equalListTail = null;
  let greaterListHead = null;
  let greaterListTail = null;

  let node = head;
  while (node !== null) {
    if (node.value < k) {
      [smallerListHead, smallerListTail] = growLinkedList(smallerListHead, smallerListTail, node);
    } else if (node.value > k) {
      [greaterListHead, greaterListTail] = growLinkedList(greaterListHead, greaterListTail, node);
    } else {
      [equalListHead, equalListTail] = growLinkedList(equalListHead, equalListTail, node);
    }

    const prevNode = node;
    node = node.next;
    prevNode.next = null;
  }

  const [firstHead, firstTail] = connectLinkedLists(smallerListHead, smallerListTail, equalListHead, equalListTail);
  const [finalHead, _] = connectLinkedLists(firstHead, firstTail, greaterListHead, greaterListTail);
  return finalHead;
}

function growLinkedList(head, tail, node) {
  let newHead = head;
  let newTail = node;

  if (newHead === null) newHead = node;
  if (tail !== null) tail.next = node;

  return [newHead, newTail];
}

function connectLinkedLists(headOne, tailOne, headTwo, tailTwo) {
  const newHead = headOne === null ? headTwo : headOne;
  const newTail = tailTwo === null ? tailOne : tailTwo;

  if (tailOne !== null) tailOne.next = headTwo;

  return [newHead, newTail];
}

// This is the class of the input linked list.
class LinkedList {
  constructor(value) {
    this.value = value;
    this.next = null;
  }
}

exports.rearrangeLinkedList = rearrangeLinkedList;

```