6247. Remove Nodes From Linked List

My Submissions (/contest/weekly-contest-321/problems/remove-nodes-from-linked-list/submissions/) Back to Contest (/contest/weekly-contest-321/)

User Accepted:	39
User Tried:	40
Total Accepted:	39
Total Submissions:	41
Difficulty:	Medium

You are given the `head` of a linked list.

Remove every node which has a node with a **strictly greater** value anywhere to the right side of it.

Return *the* `head` *of the modified linked list.*

Example 1:


```
Input: head = [5,2,13,3,8]
Output: [13,8]
Explanation: The nodes that should be removed are 5, 2 and 3.
- Node 13 is to the right of node 5.
- Node 13 is to the right of node 2.
- Node 8 is to the right of node 3.
```

Example 2:

```
Input: head = [1,1,1,1]
Output: [1,1,1,1]
Explanation: Every node has value 1, so no nodes are removed.
```

Constraints:

- The number of the nodes in the given list is in the range `[1, 10^5]`.
- `1 <= Node.val <= 10^5`

JavaScript

```
const getAllData = (list) => {
    let res = [];
    let current = list;
    while (current) {
        res.push(current.val);
        current = current.next;
    }
    return res;
};

const createL = (a) => {
    let tmp, node = null, n = a.length;
    for (let i = n - 1; ~i; i--) {
        if (!node) {
            node = new ListNode(a[i]);
        } else {
            tmp = new ListNode(a[i]);
            tmp.next = node;
            node = tmp;
        }
    }
    return node;
};

const removeNodes = (head) => {
    let a = getAllData(head), n = a.length, res = [], max = Array(n).fill(0);
    max[n - 1] = a[n - 1];
    for (let i = n - 2; i >= 0; i--) max[i] = Math.max(max[i + 1], a[i]);
    for (let i = 0; i < n; i++) {
        if (max[i] <= a[i]) res.push(a[i]);
    }
    return createL(res);
};
```

Custom Testcase | Use Example Testcases

Run | Submit

Submission Result: Accepted (/submissions/detail/850416444/) | More Details > (/submissions/detail/850416444/)

Share your acceptance!