



6013. Merge Nodes in Between Zeros

[My Submissions \(/contest/weekly-contest-281/problems/merge-nodes-in-between-zeros/submissions/\)](#)
[Back to Contest \(/contest/weekly-contest-281/\)](#)

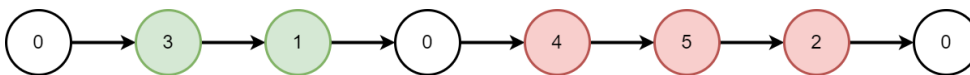
You are given the **head** of a linked list, which contains a series of integers **separated** by 0 's. The **beginning** and **end** of the linked list will have `Node.val == 0`.

For **every** two consecutive 0 's, **merge** all the nodes lying in between them into a single node whose value is the **sum** of all the merged nodes. The modified list should not contain any 0 's.

Return *the head of the modified linked list*.

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Medium

Example 1:



Input: head = [0,3,1,0,4,5,2,0]

Output: [4,11]

Explanation:

The above figure represents the given linked list. The modified list contains

- The sum of the nodes marked in green: $3 + 1 = 4$.
- The sum of the nodes marked in red: $4 + 5 + 2 = 11$.

Example 2:



Input: head = [0,1,0,3,0,2,2,0]

Output: [1,3,4]

Explanation:

The above figure represents the given linked list. The modified list contains

- The sum of the nodes marked in green: $1 = 1$.
- The sum of the nodes marked in red: $3 = 3$.
- The sum of the nodes marked in yellow: $2 + 2 = 4$.

Constraints:

- The number of nodes in the list is in the range $[3, 2 * 10^5]$.
- $0 \leq \text{Node.val} \leq 1000$
- There are **no** two consecutive nodes with `Node.val == 0`.
- The **beginning** and **end** of the linked list have `Node.val == 0`.

JavaScript



```

1 const getAllData = (list) => {
2   let res = [];
3   let current = list;
4   while (current) {
5     res.push(current.val);
6     current = current.next;

```

```
7     }
8     return res;
9 };
10
11 ▾ const createL = (arr) => {
12     let tmp, node = null;
13     let n = arr.length;
14 ▾   for (let i = n - 1; ~i; i--) {
15 ▾       if (!node) {
16           node = new ListNode(arr[i]);
17 ▾       } else {
18           tmp = new ListNode(arr[i]);
19           tmp.next = node;
20           node = tmp;
21       }
22   }
23   return node;
24 };
25
26 ▾ const mergeNodes = (head) => {
27     let a = getAllData(head), res = [], sum = 0;
28 ▾   for (const x of a) {
29 ▾       if (x == 0) {
30           if (sum != 0) res.push(sum);
31           sum = 0;
32 ▾       } else {
33           sum += x;
34       }
35   }
36   return createL(res);
37 };
```

☐ Custom Testcase[Use Example Testcases](#)[Run](#)[Submit](#)

Submission Result: **Accepted** (/submissions/detail/644939615/) ?

[More Details > \(/submissions/detail/644939615/\)](#)

Share your acceptance!

Copyright © 2022 LeetCode

[Help Center \(/support\)](#) | [Jobs \(/jobs\)](#) | [Bug Bounty \(/bugbounty\)](#) | [Online Interview \(/interview/\)](#) | [Students \(/student\)](#) | [Terms \(/terms\)](#) |

[Privacy Policy \(/privacy\)](#)

 [United States \(/region\)](#)