

Leetcode 1669:

1669. Merge In Between Linked Lists

Medium | Topics: Lists, Companies: Amazon | Hint

You are given two linked lists: `list1` and `list2` of sizes `n` and `m` respectively.

Remove `list1`'s nodes from the `a`th node to the `b`th node, and put `list2` in their place.

The blue edges and nodes in the following figure indicate the result:

Build the result list and return its head.

Example 1:

Input: `list1 = [10,1,13,6,9,5], a = 3, b = 4, list2 = [1000000,1000001,1000002]`
Output: `[10,1,13,1000000,1000001,1000002,5]`
Explanation: We remove the nodes 3 and 4 and put the entire `list2` in their place. The blue edges and nodes in the above figure indicate the result.

Example 2:

Solution:

```
1 /**
2  * Definition for singly-linked list.
3  * struct ListNode {
4  *     int val;
5  *     struct ListNode *next;
6  * };
7  */
8
9 struct ListNode* mergeInBetween(
10     struct ListNode* list1,
11     int a,
12     int b,
13     struct ListNode* list2
14 ) {
15     struct ListNode* prevA = list1;
16     struct ListNode* afterB = list1;
17
18     // Move prevA to node before index a
19     for (int i = 0; i < a - 1; i++) {
20         prevA = prevA->next;
21     }
22
23     // Move afterB to node after index b
24     afterB = prevA;
25     for (int i = 0; i < b - a + 2; i++) {
26         afterB = afterB->next;
27     }
28
29     // Connect prevA to list2
30     prevA->next = list2;
31
32     // Find tail of list2
33     struct ListNode* tail = list2;
34     while (tail->next != NULL) {
35         tail = tail->next;
36     }
37
38     // Connect tail of list2 to afterB
39     tail->next = afterB;
40
41     return list1;
42 }
```

Test Case:

SavedLn 1, Col 1

Testcase | Test Result

Accepted Runtime: 3 ms

☒ Case 1 ☒ Case 2

Input

list1 =
[10,1,13,6,9,5]

a =
3

b =
4

list2 =
[1000000,1000001,1000002]

Output

[10,1,13,1000000,1000001,1000002,5]

Expected

[10,1,13,1000000,1000001,1000002,5]

[Contribute a testcase](#)