LEETCODE PROBLEM – 1669

Description | Editorial | Solutions | Submissions

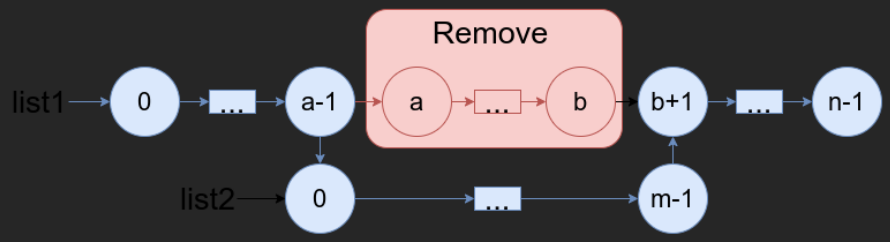
1669. Merge In Between Linked Lists

Medium | Topics | Companies | 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.

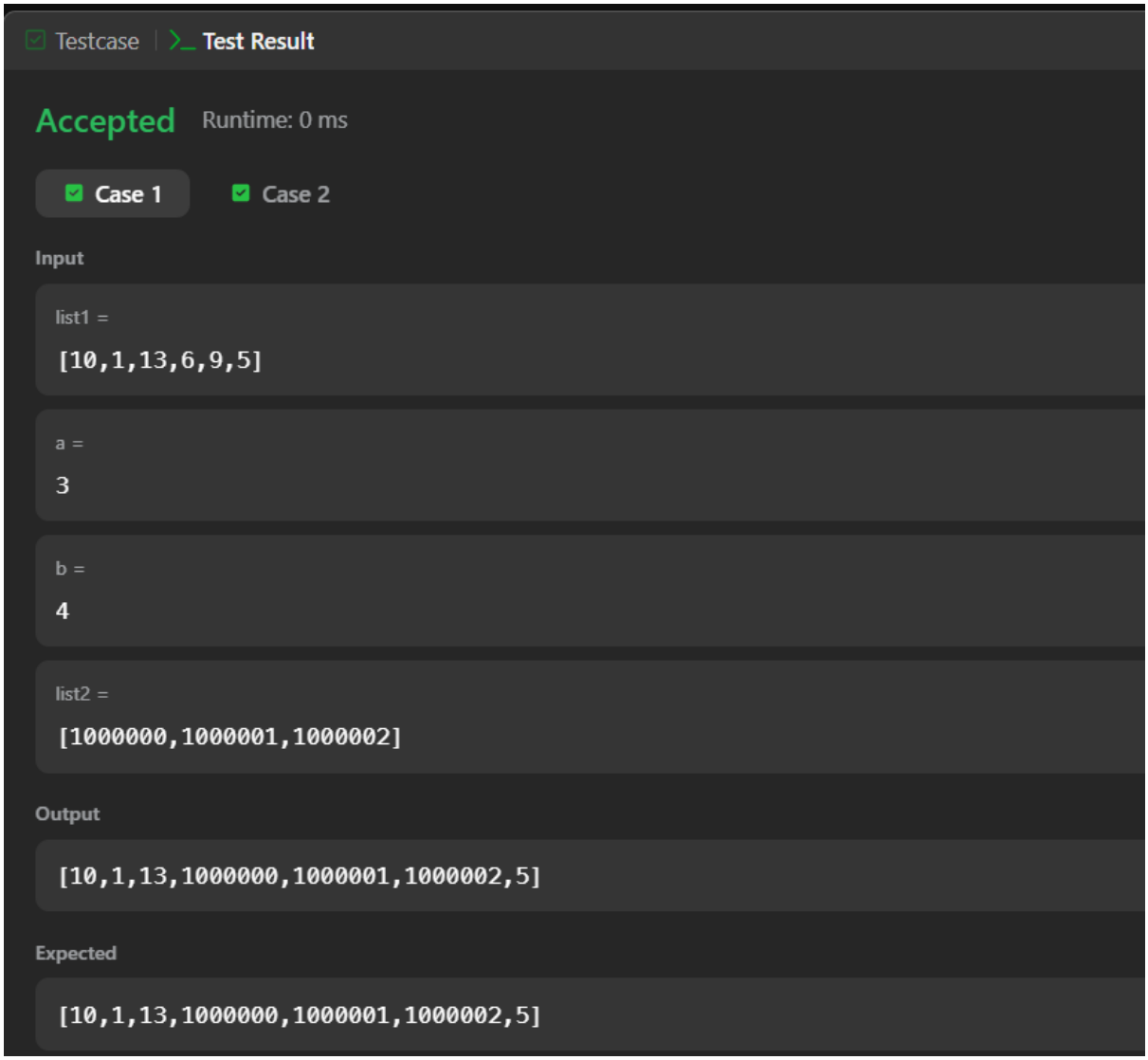
The blue edges and nodes in the following figure indicate the result:



Build the result list and return its head.

C | Auto

```c
struct ListNode* mergeInBetween(struct ListNode* list1, int a, int b, struct ListNode* list2) {
    struct ListNode *prevA = list1;
    struct ListNode *afterB = list1;

    for (int i = 0; i < a - 1; i++)
        prevA = prevA->next;

    for (int i = 0; i <= b; i++)
        afterB = afterB->next;

    prevA->next = list2;

    while (list2->next != NULL)
        list2 = list2->next;

    list2->next = afterB;

    return list1;
}
```


Testcase
Test Result
Accepted
Runtime: 0 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]