1669. Merge In Between Linked Lists

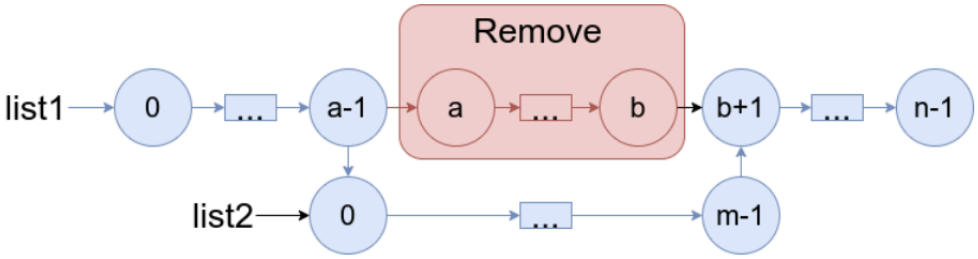
Solved

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.

```
/**
* Definition for singly-linked list.
* public class ListNode {
*     public var val: Int
*     public var next: ListNode?
*     public init() { self.val = 0; self.next = nil; }
*     public init(_ val: Int) { self.val = val; self.next = nil; }
*     public init(_ val: Int, _ next: ListNode?) { self.val = val;
self.next = next; }
* }
*/
class Solution {
   func mergeInBetween(_ list1: ListNode?, _ a: Int, _ b: Int, _
list2: ListNode?) -> ListNode? {
        var start: ListNode?
        var end: ListNode?

        var count = 0
        var temp = list1

        while(temp != nil) {
            count = count + 1
            // print(temp?.val)
            // print(count)

            // print("=======")
```

```
            if(count == a) {
                start = temp
            }
            if(count == b){
                end = temp?.next?.next
                break
            }


            temp = temp?.next
        }

        // print(start?.val)
        // print(end?.val)

        var list2end = list2

        while(list2end?.next != nil){
            list2end = list2end?.next
        }

        start?.next = list2
        list2end?.next = end

        // print("list2end. ",list2end?.val)

        return list1
    }
}
```