

# 0021 - Merge Two Sorted Lists



LinkedList



LinkedList



Timeout



Code:

```
# Definition for singly-linked list.
class ListNode(object):
    def __init__(self, val=0, next=None):
        self.val = val
        self.next = next

class Solution(object):
    def mergeTwoLists(self, list1, list2):
        """
        :type list1: Optional[ListNode]
        :type list2: Optional[ListNode]
        :rtype: Optional[ListNode]
        """

        head = ListNode()
        current = head

        while list1 != None and list2 != None:
            if list1.val <= list2.val:
                current.next = list1
```

```
temp = list1.next
current = list1
list1 = temp
else:
    current.next = list2
    temp = list2.next
    current = list2
    list2 = temp

if list1 != None:
    current.next = list1
if list2 != None:
    current.next = list2

return head.next
```

Diagram illustrating the structure of a linked list node (ListNode) and its pointer:

```
graph LR
    subgraph Node [ListNode]
        direction TB
        data[Data Field]
        next[Next Pointer]
    end
    Node --> NextNode[ListNode]
```

The diagram shows a `ListNode` structure with a `data` field and a `next` pointer. The `next` pointer points to another `ListNode` object, indicating a linked list structure.

- list1 list2 list next list1
- list1 list1 list1.next
- list next list2 list2 list2.next
- list1 list2 None (null)
- list1 list2 LinkedList Node ListNode None (null)

LinkedList

Revision #2

Created 3 July 2024 08:19:17 by Nesquate

Updated 8 July 2024 03:17:52 by Nesquate