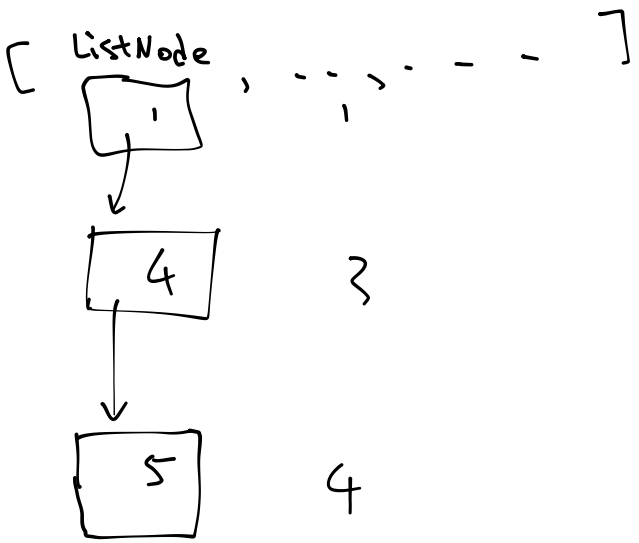
23. Merge k Sorted Lists

Hard 7770 365 Add to List Share

You are given an array of `k` linked-lists `lists`, each linked-list is sorted in ascending order.

Merge all the linked-lists into one sorted linked-list and return it.



Example 1:

```
Input: lists = [[1,4,5],[1,3,4],[2,6]]
Output: [1,1,2,3,4,4,5,6]
Explanation: The linked-lists are:
[
  1->4->5,
  1->3->4,
  2->6
]
merging them into one sorted list:
1->1->2->3->4->4->5->6
```

Example 2:

```
Input: lists = []
Output: []
```

Example 3:

```
Input: lists = [[]]
Output: []
```

A. Use mergeTwo lists as helper function to merge every possible pairs.

```python
# Definition for singly-linked list.
# class ListNode:
#     def __init__(self, val=0, next=None):
#         self.val = val
#         self.next = next
class Solution:
    def mergeKLists(self, lists: List[ListNode]) -> ListNode:
        if len(lists)==0:
            return None

        if len(lists)==1 and lists[0]==None:
            return None
        for i in range(1,len(lists)):
            lists[0]=self.mergeTwoLists(lists[0],lists[i])

        return lists[0]


    def mergeTwoLists(self,l1: ListNode, l2: ListNode) -> ListNode:

        dummy_head = ListNode()
        curr = dummy_head
        while l1 and l2:

            if l1.val <= l2.val:
                v=l1.val
                l1=l1.next
            else:
                v=l2.val
                l2=l2.next
            curr.next = ListNode(v)
            curr=curr.next

        if l1:

            curr.next=l1
```

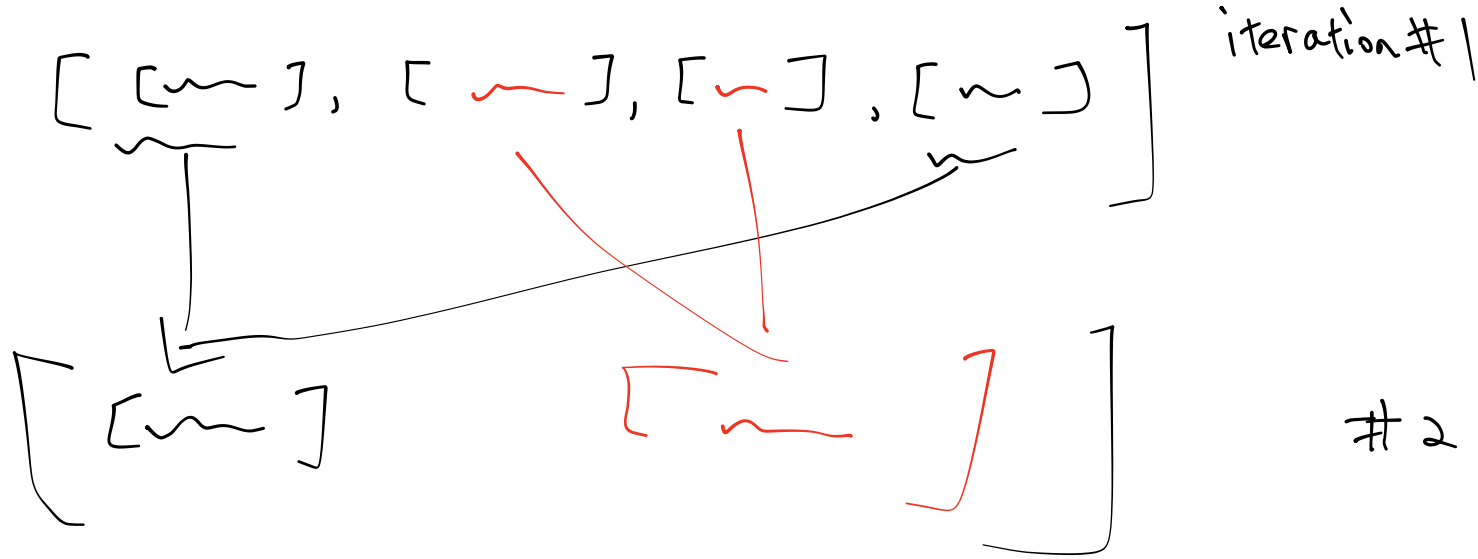
拿 [0] 与后面的 list (update each time) 一个与一个 list merge.

Bug: 忘了

B. Same idea, Divide & Conquer.

For each loop,

merge every pair



Shrink the array into half!

After each iteration. ∴ faster

```python
# Definition for singly-linked list.
# class ListNode:
#     def __init__(self, val=0, next=None):
#         self.val = val
#         self.next = next
class Solution:
    def mergeKLists(self, lists: List[ListNode]) -> ListNode:
        l=len(lists)
        if l==0:
            return None

        if l==1 and lists[0]==None:
            return None
        start = 0
        end = l-1
        while end>0:
            i=0
            j=end
            while(i<j):
                lists[i]=self.mergeTwoLists(lists[i],lists[j])
                i+=1
                j-=1
                if i>=j: # one round done, array shinked into half, will jump out of the current loop
                    end=j # update the new tail, so the outer loop runs again

        return lists[0]
```

end 是我们会 apply merge 的结尾部分