题目介绍[1]

给你一个链表数组，每个链表都已经按升序排列。 请你将所有链表合并到一个升序链表中，返回合并后的链表。

示例 1

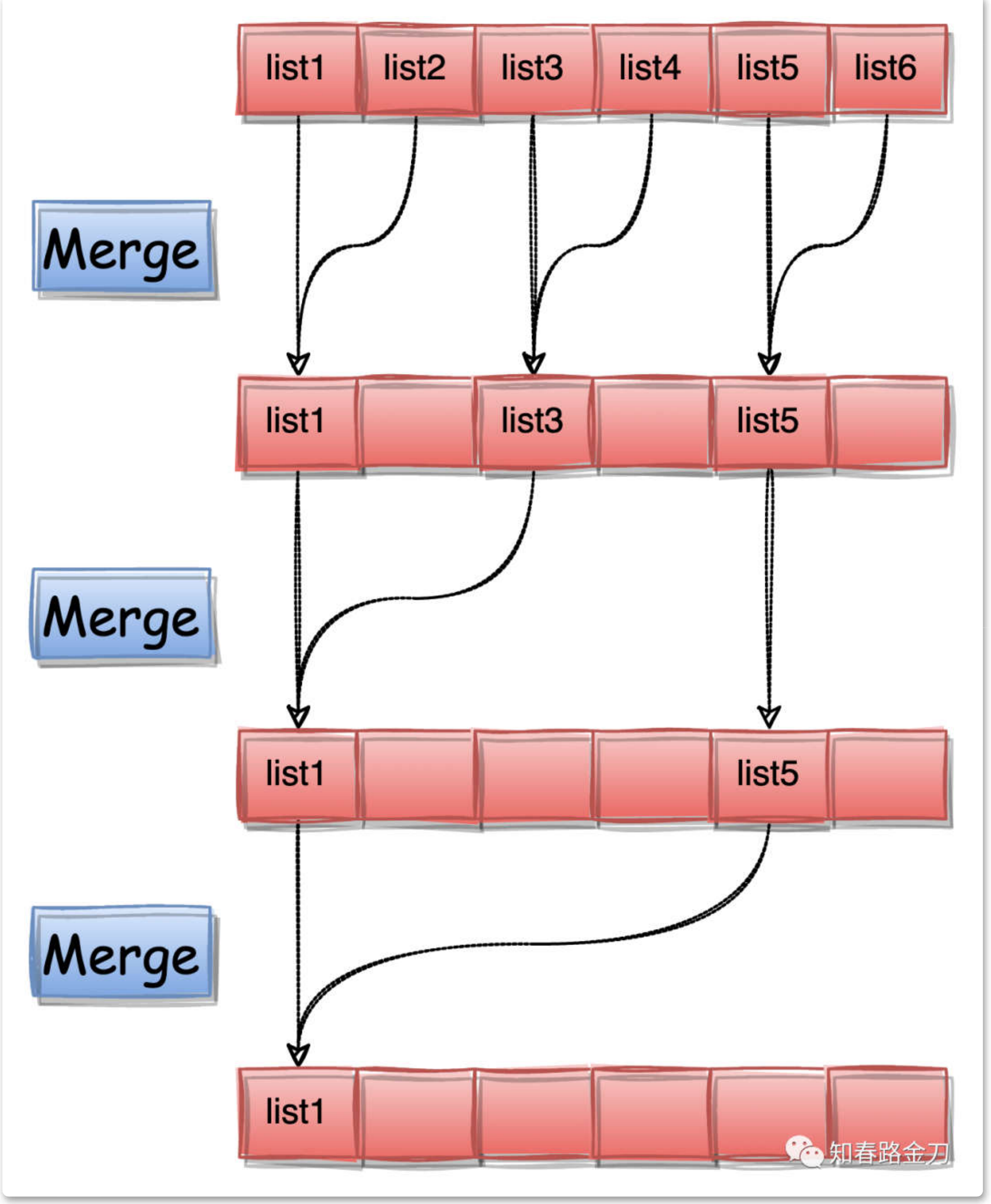
> 输入：lists = [[1,4,5],[1,3,4],[2,6]]
>
> 输出：[1,1,2,3,4,4,5,6]
>
> 解释：链表数组如下： [1->4->5, 1->3->4, 2->6] 将它们合并到一个有序链表中得到。 1->1->2->3->4->4->5->6

题目解答

这道题的解答依赖 **合并两个有序链表**，如果有不知道如何合并两个有序链表或没有思路的同学们，可以先学习一下。

两两合并(分而治之)

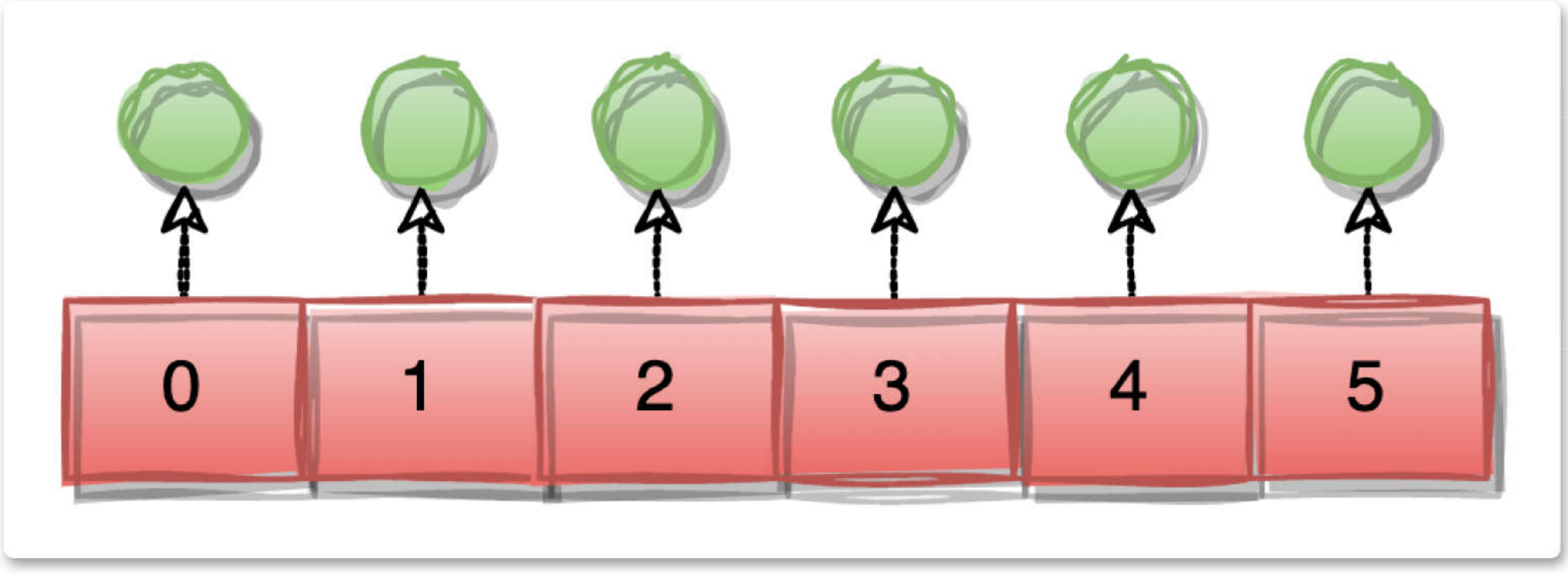
思路和算法



用分治的方法进行合并，将多个链表配对并将同一对中的链表进行合并，重复这一过程，直到我们得到了最终的有序链表。

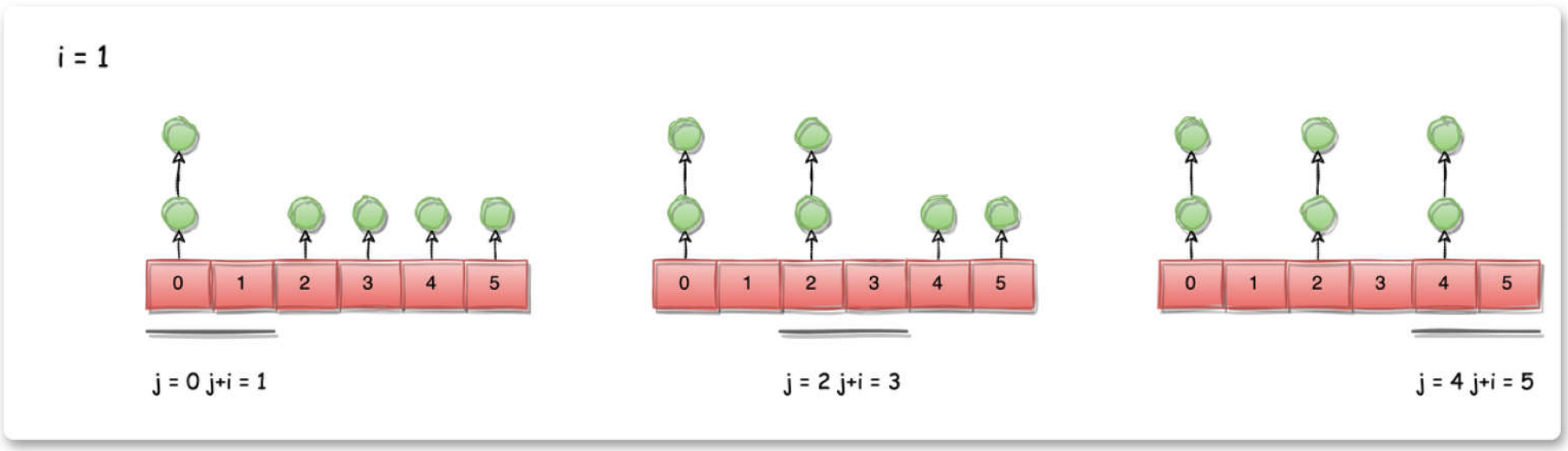
「初始状态」

初始状态一共有六个链表。



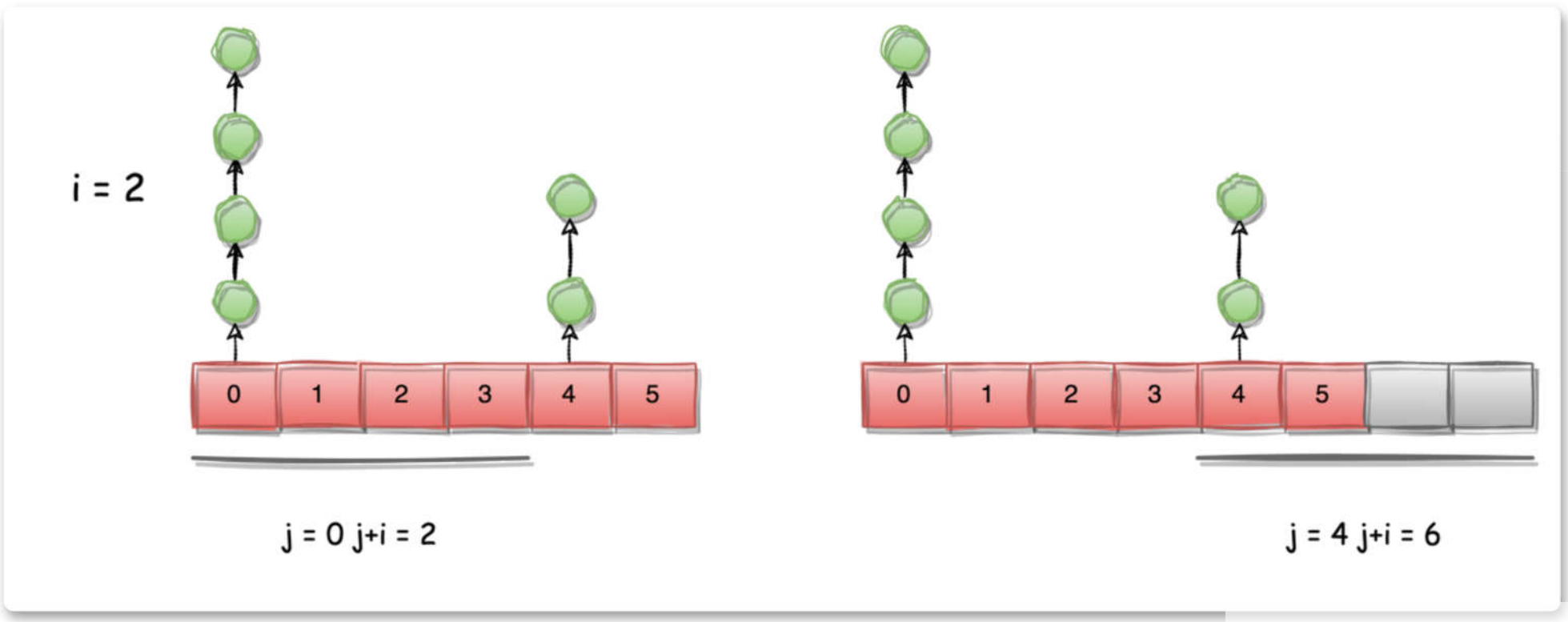
「第一次迭代」

依次对链表 {j=0, j+i=1} ，{j=2, j+i=3}，{j=4, j+i=5} 中的链表进行合并，合并完之后我们还剩下 6/2=3 个链表需要继续合并。



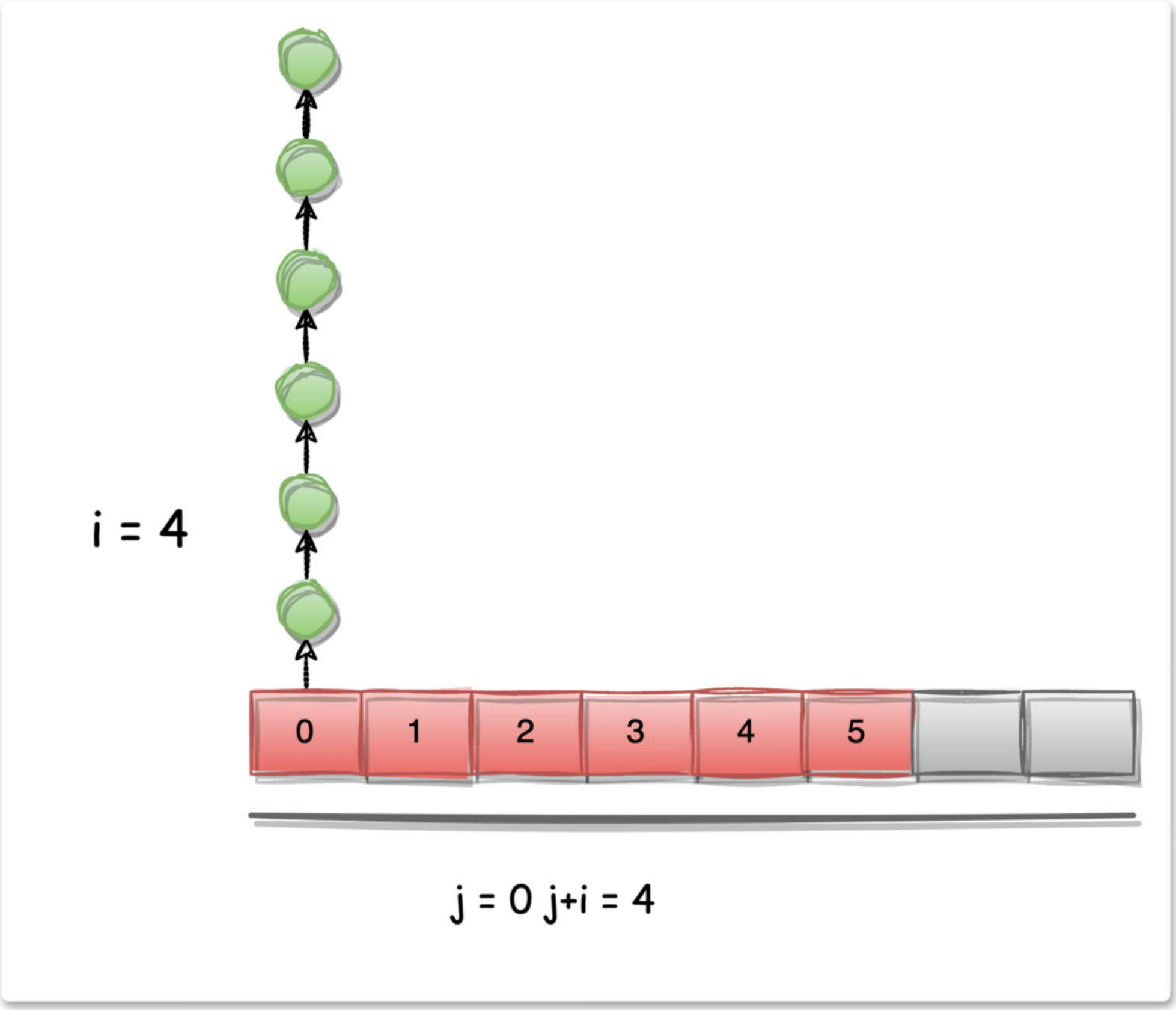
「第二次迭代」

依次对链表 {j=0, j+i=2} ，{j=4, j+i=6} 中的链表进行合并，其中灰色节点并没有链表，也没有对链表对 {j=4, j+i=6} 进行合并，这里只是画了出来方便讲解。最终还剩下两条链表。



「第三次迭代」

最后对链表 $\{j=0, j+i=4\}$ 中的链表进行合并，最终只剩下一条链表，这条链表就是我们的答案。



代码实现

```
/**
 * Definition for singly-linked list.
 * public class ListNode {
 *     int val;
 *     ListNode next;
 *     ListNode() {}
 *     ListNode(int val) { this.val = val; }
 *     ListNode(int val, ListNode next) { this.val = val; this.next = next; }
 * }
 */
class Solution {
    public ListNode mergeKLists(ListNode[] lists) {
        if (lists == null || lists.length == 0) {
            return null;
        }
        for (int i = 1; i < lists.length; i = i*2) {
            for (int j = 0; j + i < lists.length; j = j + i*2) {
                lists[j] = mergeTwoLists(lists[j], lists[j+i]);
            }
        }
        return lists[0];
    }
    ListNode mergeTwoLists(ListNode list1, ListNode list2) {
        ListNode dump = new ListNode();
        ListNode head = dump;
        while (list1 != null && list2 != null) {
            if (list1.val < list2.val) {
                head.next = list1;
                list1 = list1.next;
            } else {
                head.next = list2;
                list2 = list2.next;
            }
            head = head.next;
        }
        if (list1 != null) {
            head.next = list1;
        } else {
            head.next = list2;
        }
        return dump.next;
    }
}
```

复杂度分析

- 时间复杂度：每一轮都需要对 N 个节点进行合并，总共需要合并 logN 轮，所以时间复杂度为 O(NlogN) 。
- 空间复杂度：O(1) 。

参考资料

[1] **原题链接:**
https://leetcode-cn.com/problems/merge-k-sorted-lists/