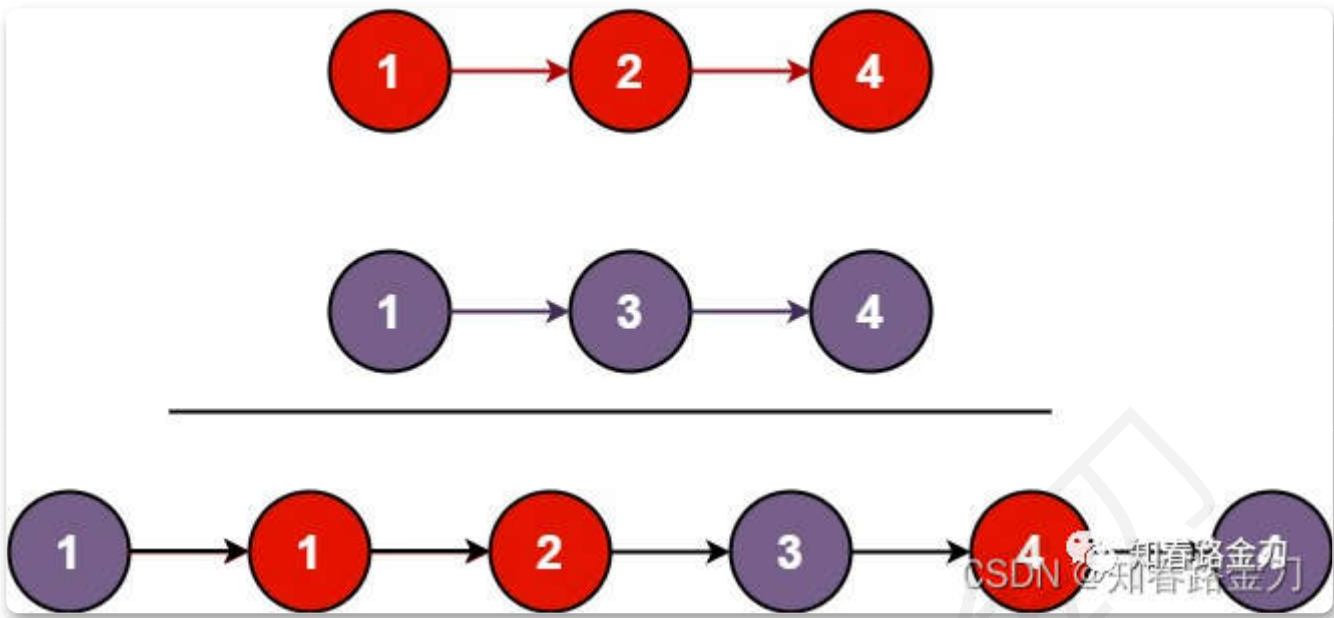


PS: 这道题的解法是很多中等和困难难度题的基础，希望大家能熟练掌握。

题目介绍^[1]

将两个升序链表合并为一个新的升序链表并返回。新链表是通过拼接给定的两个链表的所有节点组成的。

示例 1



输入：l1 = [1,2,4], l2 = [1,3,4]

输出：[1,1,2,3,4,4]

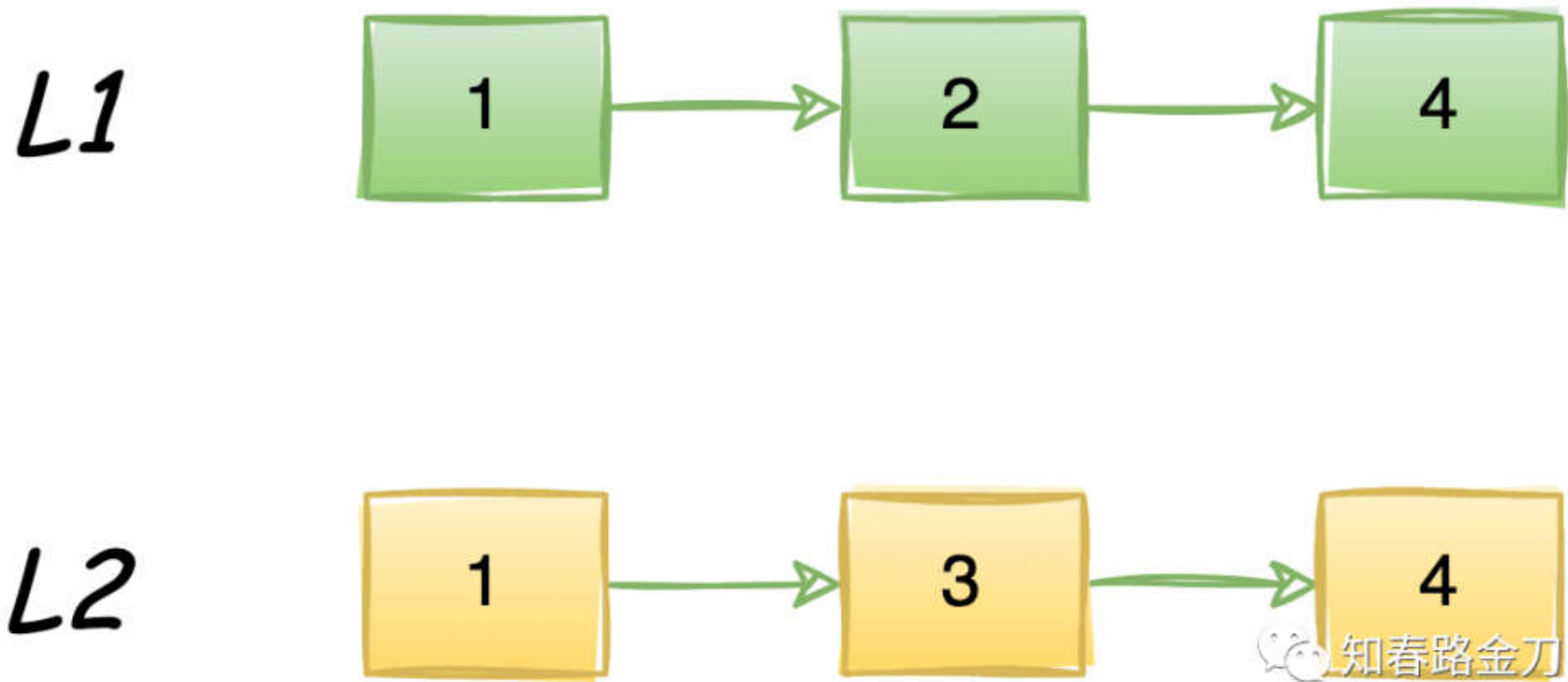
题目解答

方法一：迭代

思路和算法

迭代的思路显而易见，我们可以持续判断这两个链表头结点 l1 和 l2 是否为 null，如果都不为 null 时，去比较 l1 和 l2 头节点的值，把较小值的头节点拼接到新的链表里去。

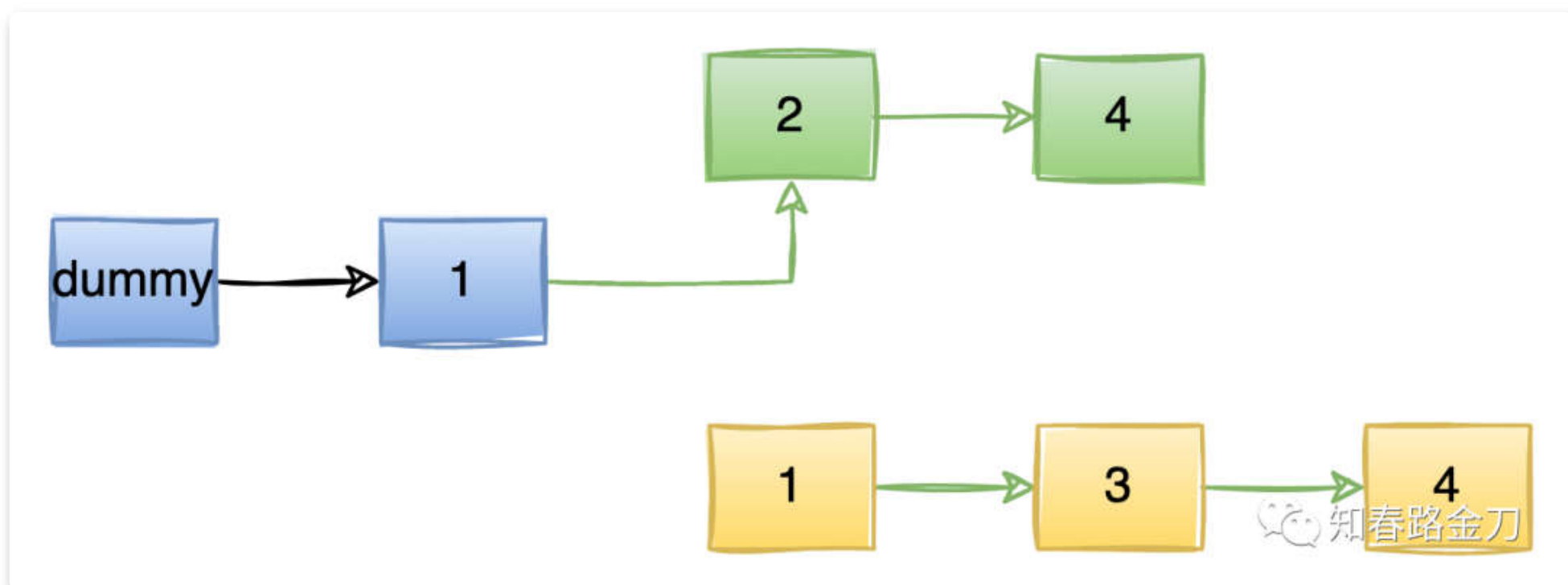
「初始状态」



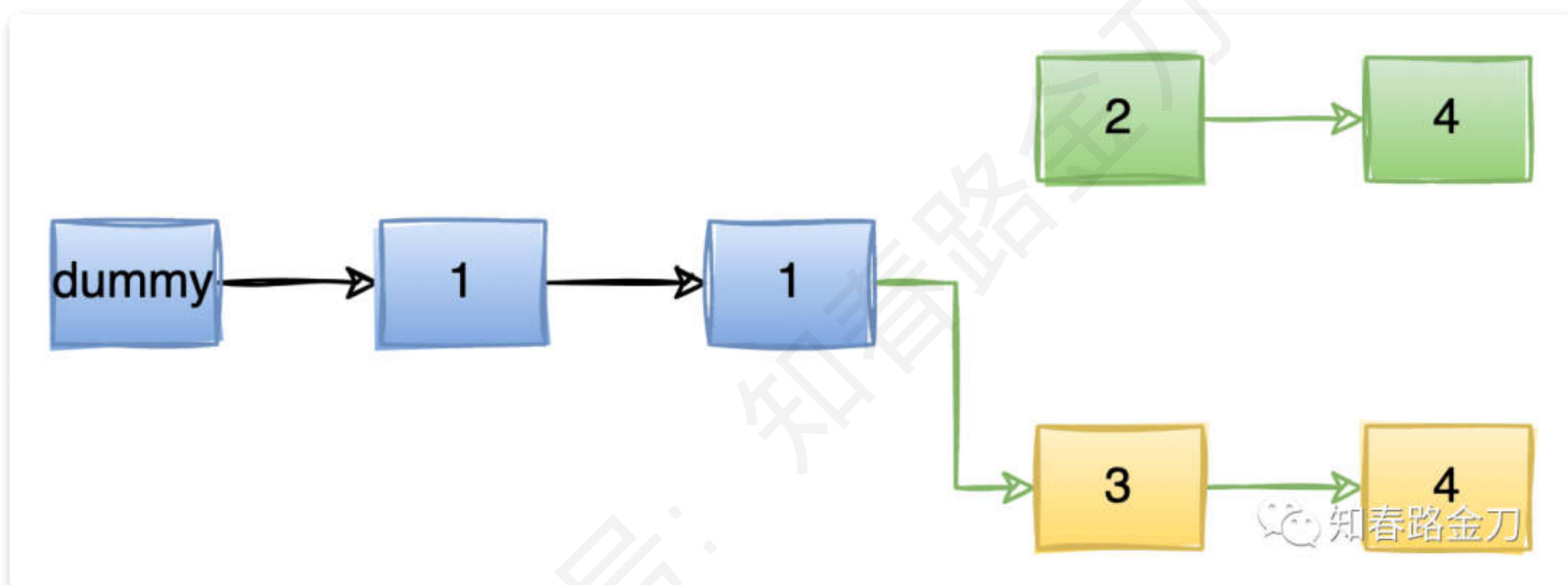
微信搜一搜

知春路金刀

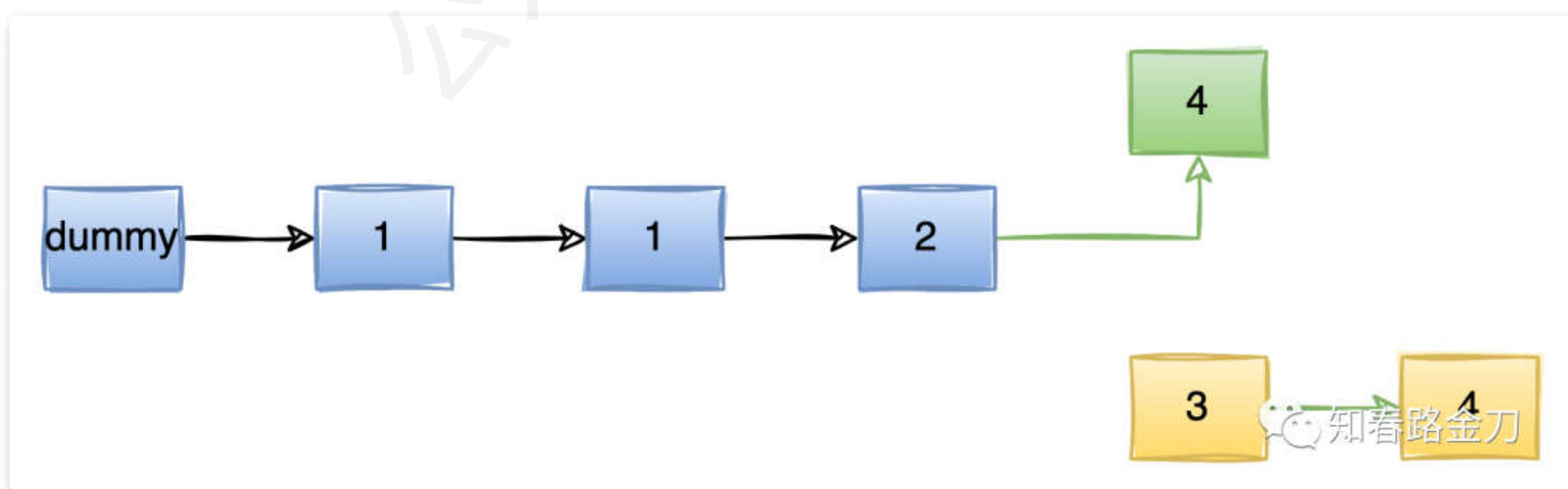
「第一次迭代」



「第二次迭代」



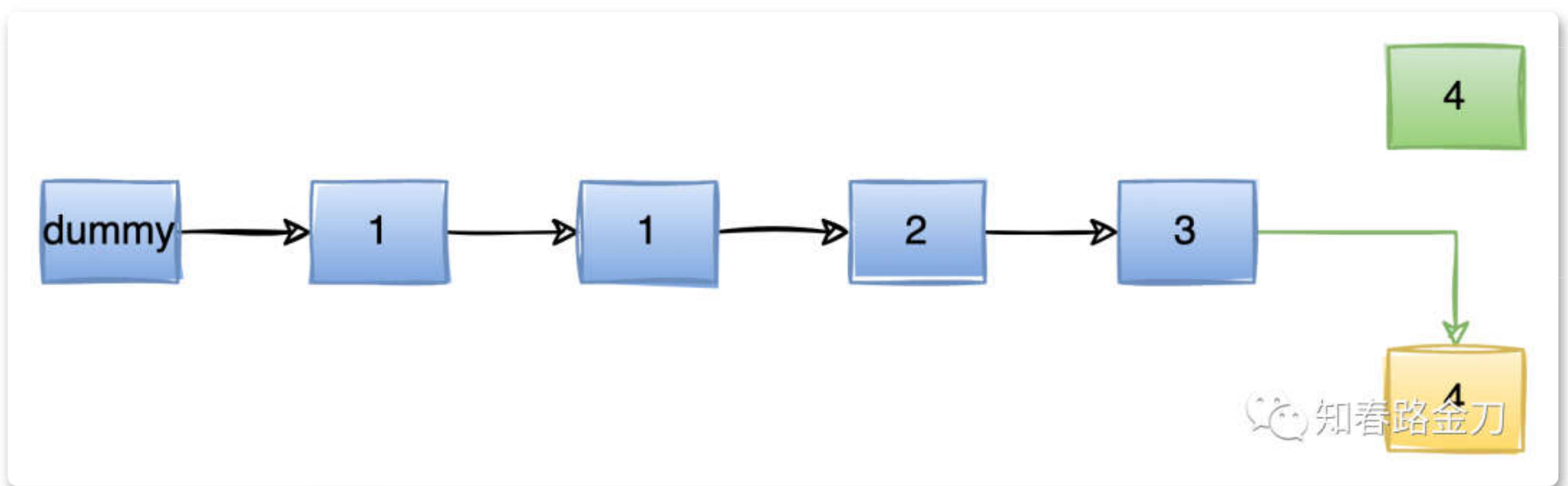
「第三次迭代」



微信搜一搜

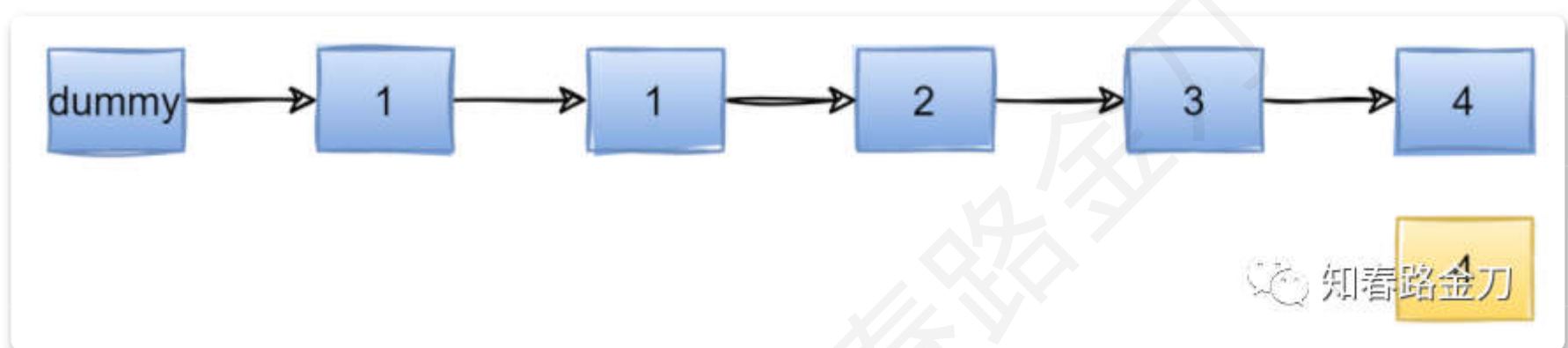
知春路金刀

「第四次迭代」



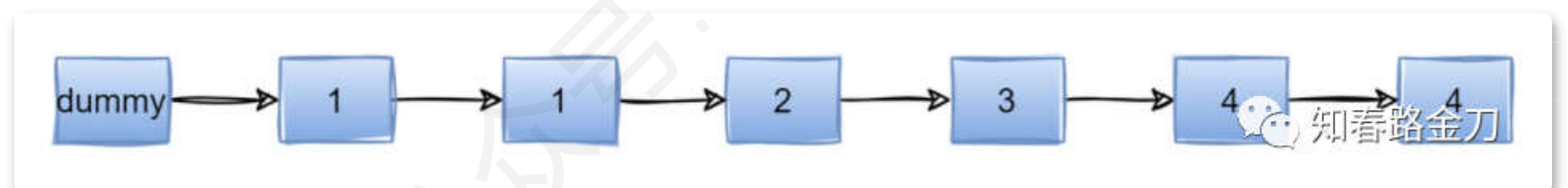
「第五次迭代」

这时终止循环，l2 链表只剩下一个节点。



「拼接剩余链表」

把 l2 直接拼接到新链表中。



微信搜一搜

知春路金刀

代码实现

```
class Solution {  
    public ListNode mergeTwoLists(ListNode l1, ListNode l2) {  
        ListNode dump = new ListNode();  
        ListNode head = dump;  
        while (l1 != null && l2 != null) {  
            if (l1.val < l2.val) {  
                head.next = l1;  
                l1 = l1.next;  
            } else {  
                head.next = l2;  
                l2 = l2.next;  
            }  
            head = head.next;  
        }  
        if (l1 != null) {  
            head.next = l1;  
        }  
        if (l2 != null) {  
            head.next = l2;  
        }  
        return dump.next;  
    }  
}
```

复杂度分析

- 时间复杂度: $O(n+m)$
- 空间复杂度: $O(1)$

参考资料

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



微信搜一搜

知春路金刀