

链表高频题目和必备技巧

前置知识：

讲解009~012-链表入门内容、讲解021-归并排序

讲解026-哈希表的使用、讲解029-排序算法的稳定性

链表类题目注意点：

- 1，如果笔试中空间要求不严格，直接使用容器来解决链表问题
- 2，如果笔试中空间要求严格、或者在面试中面试官强调空间的优化，需要使用额外空间复杂度 $O(1)$ 的方法
- 3，最常用的技巧-快慢指针
- 4，链表类题目往往都是很简单的算法问题，核心考察点也并不是算法设计，是*coding*能力
- 5，这一类问题除了多写多练没有别的应对方法

个人建议：链表类问题既然练的就是*coding*，那么不要采取空间上讨巧的方式来练习

注意：链表相关的比较难的问题是约瑟夫环问题，会在【扩展】阶段讲解，变形很多会单独出一期视频讲解

链表高频题目和必备技巧

题目 1 ：返回两个无环链表相交的第一个节点

题目 2 ：每 k 个节点一组翻转链表

题目 3 ：复制带随机指针的链表

题目 4 ：判断链表是否是回文结构。这个题的流程设计甚至是考研常用。快慢指针找中点。

题目 5 ：返回链表的第一个入环节点。快慢指针找中点。

题目 6 ：在链表上排序。要求时间复杂度 $O(n * \log n)$ ，额外空间复杂度 $O(1)$ ，还要求排序有稳定性。

注意：

这些题目往往难度标为“简单”，是因为用容器解决真的很简单

但是不用容器、实现额外空间复杂度 $O(1)$ 的方法并不轻松，包括很多提交的答案也都没有符合要求