

1.开班典礼

要面试的时候才会重视算法

算法边界：明确

高频常见题目：必须会，没有理由

数组：高级算法的载体。

链表：极少作为高级载体，刷完就完了

树：一个庞大体系，学习的重点（树、二分查找、快速排序、归并排序、回溯、动态规划）

直播：带路，让你快速了解算法到底在干什么。

35天 5天直播 划重点

直播的内容只占讲义的 1/3

忘了就再来遍。

第一轮：35天直播：

第二轮：直播结束，讲义认认真真搞一遍

第三轮：面试之前，突击抱佛脚

2.链表的学习任务：

1.理解链表的原理，验证方式：能够理解讲义2.1里题干的两个是如何构造的

2.自己实现链表的增删改查，能够处理在链表的首部、中间和尾部操作元素，基本功。

3.理解双指针的原理，理解2.4中链表的环是怎么判断的。重点是为什么step=2的时候，fast和slow一定能够相遇。

4.如何确定环的入口，两种方法，第一种是三次使用双指针。好理解，但是代码繁琐。

第二种：正统的方式，不好理解，代码好写。

3.时间复杂度和空间复杂度

1.什么是时间复杂度

简化：

第一个简化：只看执行次数，不看实际执行时间，或者说用执行的次数作为执行的时间。

第二简化：对于执行次数，只保留影响最大的因子，其他的都去掉，数学里渐近线和无限逼近（算法导论第一章）

2.面试怎么考

1.你写完之后，面试官问，你的算法时间复杂度是多少？

2.能否将复杂降低到 $O(n)$ 或者 $O(1)$ 阶。

3.能否在 $O(n)$ 的时间范围解决。

3.常见算法的时间和空间度

复杂度学习任务

1.理解时间复杂度的两次简化

2.常见的阶如何计算（讲义有练习题）

3.理解常见的空间复杂度

凡是申请的是 数组、队列、栈、hash 、集合 空间复杂度都是 $O(n)$

二维数组 必然 $O(n^2)$:动态规划

申请若干个 变量 $O(1)$

int slow,fast,cur,next;