1 2 3 4 15 19 20 34 46 94 98 124

OK 1 Two Sum 实践：hashMap知道value如何求得key？

2 Add Two Numbers 思路不难，很简单。 实践：OK的。

背3 最长不重复字符串 看你能不能想到。虽然看起来像是用动态规划，但是最长不重复字符串，以及字符串，能想到使用什么方法呢？ 实践：

4

5 最长的回文子串 因为太经典，所以记住解法即可，动态规划 实践：做一遍，最难的还是究竟怎么遍历能够最好!!真的蛮难的。

--6 ZigZag转换 这题出得真怪

--7 反转整数 要考虑的是溢出的问题。 实践：

\*10 1 1 1 1 1 1

11 装最多水 还是思想的问题。可以把收集起来。想想与暴力法相比这种方法的优势在哪里？ 实践：有思路就能做。 双指针在这种题目下能筛掉一些明显并不会为结果带来帮助的选择。否则一般情况下并不会对复杂度有帮助。

15 3 SUM 实际上是为了避免三重循环，如何减少时间复杂度的问题。你会如何做？ 两种？ 实践：有思路能做。 只有答案的方法算是正解。

排序数组在找唯一组合的时候要注意什么问题?

17 电话号码的字母组合 应该算是一种典型的题目 实践：

19 一次遍历除去链表倒数第N个结点 实践：有思路应该能做。

20 验证括号 实践：1 初始构造map的地方写得不太好，重写 2 栈空的情况希望写得更加精简。3 没有考虑到右括号没有成功进入的情况。3 看来isEmpty() 和empty()都可以。

21 两个排序链表合一

--22 生成括号 难想，考核点少先跳过。

OK默写23 合并k个排序链表 主要会写分治法的部分即可 实践：分治法，奇数偶数如何求得k,这个存在问题。 分治要求两部分内容1 一部分是mid前与mid后相对应数之差 2一部分是mid左边第一个数的下标。

31 先跳过

32 括号题先跳过

33

背34 搜索一个范围 想要记住细节非常困难，背下来背下来！！ 实践:1right=nums.size() 2res[1]=left-1 还有第二个是小于等于

42 盛的雨水量 思想最重要。 实践：还需要再做。

注意：在双指针法 的时候，要时刻注意保持的是什么?

背46 全排列 主要是记住固有的递归的做法 实践：

48 旋转数组 记住固有解法即可 实践：没什么大问题

你只要写出数组排序的部分即可。49 群组错位词 两种思路都蛮有特点，蛮典型的。都值得一看 实践：1转化为字符数组，与转化为字符串的写法格式 2HashMap里的内容取出存储到ArrayList中

//同一个Key可以对应多个吗? 能找到对应Value然后往里面添加吗？（保险起见还是先照着自己的写吧） //如何取出map中的每一个值录入一个集合？？ 思路是没什么问题。有思路不影响写出来。

<http://www.runoob.com/java/java-collections.html> 这个讲了hashmap遍历的四种方式

53 连续子序列

55 Jump Game 一道经典的动态规划题目。要记住动态规划的核心解决问题是什么！

实践：没什么问题，主要是步数的定义搞清楚就行。

56 合并区间 有点难，先跳过

先跳到94， 开始做数据结构的题目吧，考到可能性更大。

背：94 二叉树的中序遍历 可以看看迭代法是怎么做得

--96 独一无二的二叉搜索树 算是一种比较独立的公式型题目.所以个人觉得考的可能性很小 实践：

可以看一下，可以对中序遍历有更深理解98 判断是否是二叉搜索树 两种解法都要掌握， 实践：第一种解法没什么问题，第二种解法再写一下，不要写出臃肿的代码。

101 判断是否是对称二叉树

102 层序遍历二叉树

104 二叉树的最大深度

105 由前序中序构造二叉树

114 二叉树变为单向链表（应该比更简单）

121 购买股票的最佳时机 思想 实践：遇到不好定义的变量，要怎么初始化?

\*124 二叉树中和为某一值的路径 如何把路径list加入到大List中？ 如何解决用于临时记录的list数组在递归中的处理？

实践：注意，不能把集合加入大集合中，必须要重新new一个才行。

128 \* 先跳过

136 找出没有重复2次的数 实践：没问题

139 word break 还没消化完 实践： 。。我觉得还是自己的方案比较好。

HashSet方法 https://blog.csdn.net/tingzhiyi/article/details/52152487

141 循环结点 经典题目，扫一眼做一下吧 实践：

掌握循环方法142 循环结点2 有思路，了解关系就能做 实践：妙啊，第二遍没做出来

有个很大的疑问，快慢结点从哪里开始？

\*146

148 排序链表 要掌握了归并排序才可以做，有些难

152 求最大子数组乘积 比较难的动态规划， 实践：

141 142 148 160

155 min堆栈

160 求两个链表的交集 主要的难点是时间空间复杂度的限制，最终的方法更像是智力题，比较巧妙。像链表这种可用的方法不多的，基本都是从多指针来入手的 实践：会思路即可 批注：链表这种题目有了巧妙地方法就变得很简单啊。。

OK169 数组中最多的元素 巧妙的方法不强求，主要是掌握如何从hashmap中找到value最大的值，这样要记下来怎么写！！ 很重要！！ 实践：实际上还有一种 for( int count:map.values() ) 的方法。可以得到values。

OK198 打家劫舍 动态规划的题目，动态规划每次都翻车，看来要在之后弄个专题了 实践：对于转移方程有点疑问，先死记下来。思想是掌握了。

OK200 岛屿数量 深度优先DFS，这种题目最好归类好 思路其实不难的，主要是想到。实践：

206 反转链表

437 Path Sum III 看样子我们递归的题目做少了都不会做了，蛮好的题目。 树的题目说实话手感大于技巧感。因为很难描述思路。就是多做 return的返回结果的写法看样子比全局变量的写法更有技术含量啊。这题虽然做出来的，但是怎么能算easy呢?

438 根据给出的小字符串，在大字符串中找小字符串，顺序可以不同 这题实际上和之前的字典的思路是一样的。

实践：

\*448 找所有没有在数组中出现的数字 这题很难啊，easy个jb。我答案都看得一脸懵逼。

461 汉明距离 典型的异或类题目，最好做一下。 实践：OK。一个int类型数字a的二进制有几位？如何遍历?

494 很好很经典的递归题目，关于减枝操作还没完全弄懂 实践：

OK538 将BST转换为Greater Tree 一种额外空间的，一种楼主的做法，两种思路都回想一遍 .如何将二叉搜索树从大到小遍历？ 实践：

543 二叉树的直径 只要明确知道怎样路径最长，就比较好做。怎样的路径最长？ 实践：我们原本的理解存在问题。

背560 数组中加和为k的连续子数组 这是一道很典型的防止多次遍历的题目，至少先记住简单的那种缩小复杂度的方法 实践：

572 另一棵树的子树

581 ？

617 融合两个二叉树 似乎没什么问题，就是普通的递归

--621 题目表述这么复杂的题目，肯定不会考，跳过。

647 回文子字符串 动态规划！！与递归，思维过程还是复杂的。递归之后也试着做一下吧 实践：

721 珠宝和石头

56-94

62 唯一路径机器人移动

64 最小路径和矩形移动 没想到用这种方法吧？ 实践：会原理会做。

70 爬楼梯

72\* 编辑距离 我想只要能画出关系图就能够推导出来。 实践：

75 Sort Colors 还是比较简单的 一个数组中只有已知种类的字符，如何排序？ 实践：

想到的是计数，但是想不出要怎样不动用额外的空间来完成这一过程？ 或者如题目所说不动用额外空间。 一。

76\* 最小窗口滑动 应该是很重要但是掌握比较麻烦地题型，先跳过吧

78 子集合 如何创建一个和原来的集合内容一样的ArrayList ? 主要还是思路 实践：

79 word Search 典型的深度遍历机器人类题目。 实践：先试着回忆。 后半段不太熟悉，对visit[] 以及index参数的传递不够熟练。 以及可以学习一下它的思路。

84\*

85\*

207-416

207\* 有向图的 ， 不管啦

208 不顺眼先跳过

215 数组中第k大的数字 为什么能利用快排得到第k大的数字？ 实践：

--221 在字符矩阵中找到最大正方形面积 没什么代表性的题目，记住解法。

实践：

226 反转二叉树

234 回文链表 palindrome是什么意思？ 快慢指针的具体写法是什么样的？快指针最终要在哪里停下呢？慢指针一定处于中点吗? 实践：没问题。

实践：

236 最小共同祖先 这题的逻辑还是蛮复杂的，虽然代码很简单。 实践： 利用我们做递归题目的技巧，做了大半， 但是没有推导出左右两边都有的情况。 还是蛮难的。

238 除了本身之外的数组之积

239\* 滑动窗口 deque的取出，删除，查看方法？

<http://how2j.cn/k/collection/collection-linkedlist/370.html#nowhere>

这题是要掌握的。

240 在排序矩阵中找值

253 这题暂时打不开。。 ？？

--279 Perfect 平方数 数论题。相当于智力题，没有代表性 记住。。算了，也别记住了，这题考不出什么吧。。

283 移动0 思路比冒泡好上很多，你能想出来吗？ 实践：非常巧妙的方法，没想出来。不过思路最重要。

287 寻找重复数 Duplicate是什么意思？ 为了查询而降低时间复杂度可以怎么做，这个数组不是有序数组，也可以用二分法吗？ 实践：实际上这种方法是比不上快排的速度的。

297\* 树的序列化与反序列化 暂时找不到JAVA解法，先跳过

300 最长递增队列 能想到那一层的话，这题就不算太难。 实践：没问题

301\*

309难 有冷却期的买卖股票。 很难想到的方法。我觉得这题除了代码总量之外就是一道非常难的动态规划，这个场景更像是高阶的奥数题。

312\*

322 Coin Change 身为中等难度的题目还是好难。 实践：如何设定数组的初始值？遍历的过程中要顾及前面的问题么？

背337 入室窃贼 原理和奇怪的点和简单版的一致，思路清晰即可。 实践：几个问题？

为什么要优化？ 哪些地方存在重复?和动态规划比?

--338 个人认为这种找规律的题目一般不会出。

背：347 前K个频繁出现的数字 重要的题目 实践：1 涉及到map key以及value的遍历。 2 桶排序。

--394 烦人题

406

416 相同子集分割

机器人类题目： 有什么注意的固有的思路技巧？

1 记录是否重复基本都是依靠维护数组，不可能让你拿新的和之前的无数的暴力比较。

StringBuffer append

Queue offer() poll()

所有用到的工具：

数组类题目： 1直接遍历，更新min max 2 工具类，比如hashMap hashSet等等

数组以及工具类 hashMap 栈 如果判断是否重复，尝试用hashSet 3动态规划（记住要想清楚每个dp【】代表的是什么含义） 4链表常见的是双指针，快慢指针 一般排序数组都是想到使用二分 5 双指针法，有可能是两个相邻的，两个一起前进，一前一后，也有两头向中间的。然后很多时候的思想是快排思想。

分治法。

递归如果求在递归过程中累积的值的话，1一般都通过sum（全局参数）累积数量，2return

3如果往递归中把外面的数组作为参数传入，那么不存在回溯的问题。改了就是改了。因此如果一条路走到死路了，就要把之前更改的数组内容改回来。

4 有些时候恰恰要利用参数回溯的特性，使得分支相互之间不影响。

5 递归的题目在写递归方法的时候，要弄清楚这个方法究竟是做什么的。是以root参数为起点的最大值还是什么。清楚了这个之后就更容易写出左右子树的关系。更有利于写出方法。

排序以降低时间复杂度： 1 比较排序中的快排 2 桶排序， 空间换时间。

二分法是基本上最高效的查找算法。而能够使用的情况仅限于排序或者一些暗藏规律的条件。

归并排序的知识：

1、//将temp中的元素全部拷贝到原数组中 //

//之所以这样，是因为递归的每层，都相当于更新一次数组。

//前一层更新了数组，后一层才能够继续合并。 而temp[]起到的唯一的作用就是暂时记录数据，此外就没有任何作用了。

94 二叉树的中序遍历 可以看看迭代法是怎么做得

背：

你的思维存在很大的漏洞，只会死记住一种方法来做，但是根本没有想过多种方法混用，一看到用动态规划，你就只能想到用动态规划来做。但是不会想到再加上双指针。来帮助你。

背：剑指Offer（三十七）：数字在排序数组中出现的次数 重要的，典型的二分法题目。重点：1 mid如何计算？ 2 二分法如何分别得到最前和最后的。？ 比leetcode的写得好

实践：会写二分法的其中一边即可。 知道为什么要mid-1吗？ 这是因为array[mid-1]的判断也是不允许越界的。

默写： 着重默写getfrist的逻辑即可。

字符串组的全部！！

**反转链表（死记）d**

**平衡二叉树（这题目原本的方法有误） 实践：能不能背下来，真的是很好的一段代码，堪称艺术。 此外还有一种后序遍历的方法也很好。**

默写：后面的部分有点问题。

默写23 合并k个排序链表 主要会写分治法的部分即可 实践：分治法，奇数偶数如何求得k,这个存在问题。 分治要求两部分内容1 一部分是mid前与mid后相对应数之差 2一部分是mid左边第一个数的下标。 记住那两个点就可以顺利写出来？

默写：下标小于n/2 的即为右半部分。 (n+1)/2 即为前后两边的间隔。

背34 搜索一个范围 想要记住细节非常困难，背下来背下来！！ 实践:1right=nums.size() 2res[1]=left-1 还有第二个是小于等于

背46 全排列 主要是记住固有的递归的做法 实践：

背560 数组中加和为k的连续子数组 这是一道很典型的防止多次遍历的题目，至少先记住简单的那种缩小复杂度的方法 实践：

默认：双指针遍历法如何以某个为起点的形式完成?

第四题重点 ：但是还是别记标准答案了，这题太难了。按照hashmap的方法做就是了。

默认：

归并排序：主要记住思想

既然Arrays 没有reverse方法，那么要怎么反转字符串？ 二分法或者创建数组，用Collections <https://blog.csdn.net/neweastsun/article/details/79935638>

面试算法题目的时候一个技巧是先弄清楚题目要做什么，此外要注意规范，先确认给的数据有没有问题。

背 二分法总结

http://www.cnblogs.com/grandyang/p/6854825.html

我们稳定查找与目标值完全相等的数的写法要怎么写?

查找第一个不小于目标的数， 查找最后一个小于目标值的数？

查找第一个大于目标值的数，朝朝最后一个不大于目标值的数？

背337 入室窃贼 原理和奇怪的点和简单版的一致，思路清晰即可。 实践：几个问题？

为什么要优化？ 哪些地方存在重复?和动态规划比?

背：347 前K个频繁出现的数字 重要的题目 实践：1 涉及到map key以及value的遍历。 2 桶排序。

背3 最长不重复字符串 看你能不能想到。虽然看起来像是用动态规划，但是最长不重复字符串，以及字符串，能想到使用什么方法呢？ 实践：

重点看： 剑指offer路径和为某一值。 ArrayList怎么remove最后一个？

如何写快慢指针？ 数组中的遍历问题 回忆快慢指针的第二题的做法？

Entry的写法格式？ 461 汉明距离 494 记录中间值。 融合两个二叉树。

珠宝和石头

75Sort Colors 快排思想的体现 78子集合

难但重要 146 排序链表 438 找子字符串（考的可能较小） 543（考到的可能性不高）