

基础数据结构

数组、字符串

链表

栈

队列

双端队列

树

高级数据结构

优先队列

图

前缀树

算法思想

递归

回溯

动态规划

二分搜索

贪婪算法

[LeetCode面试高频题目](#)

以下内容为300分钟搞定算法面试的学习笔记。总结在公众号“跟小新一起来编程”。



基础数据结构

数组、字符串

链表

特点：数据元素个数不确定，经常需要删除或者插入元素而不是经常访问某个元素。

应用：

- 快慢指针（有时候三个指针）：链表翻转、寻找链表倒数第k个元素、寻找链表中间位置元素、判断链表是否有环。
- 构建一个虚假的链表头，优点是不需要判断链表是否为空，删除和插入简单了，返回时返回链表头的下一个元素；两个排序链表进行整合；将链表的分离，前半部分是奇数，后半部分是偶数。

栈

特点：后进先出。

基本思想：只关心最后一次的操作，处理完上一次的操作，能在 $O(1)$ 的时间内查找到更新前的一次操作。

LeetCode：20有效的括号；

LeetCode：739每日温度。

队列

特点：先进先出

应用：广度优先遍历

双端队列

队列的头尾两端能在 $O(1)$ 的时间内进行数据的查看、添加和删除。

常用场景：实现一个长度动态变化的窗口或连续区间

LeetCode：239 滑动窗口的最大值

树

树的问题考查递归算法的熟练程度。

常考的树：普通的二叉树；平衡二叉树；完全二叉树；二叉搜索树；四叉树

遍历：前序遍历（树的创建）、中序遍历（如果是二叉搜索树，遍历结果是排序好的，LeetCode 230题）、后序遍历（LeetCode 250题）

高级数据结构

优先队列

与普通队列的区别：保证每次取出的元素是队列中优先级最高的，优先级自定义。

最常用的场景：从杂乱无章的数据中按照一定的顺序筛选数据。例如：一个数组的前K大的数据。

本质：本质是一个二叉堆，利用一个数组结构来实现完全二叉树。

LeetCode：347 前K个高频元素

图

最基本知识点：度、有向图、无向图、连通图。

图的算法：图的遍历——深度优先与广度优先；环的检测；拓扑排序；最短路径算法（Dijkstra、Floyd）；连通性相关——求解孤岛数量、旅行商问题、图的着色。

必须掌握的算法：

1.图的存储和表达方式：邻接矩阵、邻接链表

2.图的遍历：深度优先和广度优先（重要）

基于图的遍历的算法：二分图的检测、环的检测

3.最短路径

4.并查集

LeetCode：785 判断二分图

前缀树

前缀树叶称字典树，例如：给定一系列构成字典的字符串，需要在字典中找出所有以“ABC”开头的字符串。使用前缀树。

LeetCode：212 单词搜索2

算法思想

递归

递归的基本性质：函数调用本身，把大规模问题不断变小，在进行推导的过程。

特点：可以将一个看似复杂的问题变得简洁和易于理解。经典问题：汉诺塔问题

算法思想：1.要懂得如何将一个问题的规模变小。

2.在利用从小规模问题中得出的结果结合当前值或者情况，得出最终的结果

通俗理解：1.把要实现的递归函数看出已经实现好的。

2.直接利用解决一些子问题

3.思考如何根据子问题的解以及当前面对的情况得出答案

```
1  function fn(n)
2  {
3      //第一步：判断输入或者状态是否非法
4      if(input/state is invalid)
5      {
6          return;
7      }
8
9      //第二步：判断递归是否应当结束？都是处理递归定义的初始情况
10     if(match condition)
11     {
12         return some value;
13     }
14
15     //第三步：缩小问题规模
16     result=fn (n1) ;
17
18     //第四步：整合结果
19     return combine (result1, result2);
20 }
```

LeetCode 91 解码方法

LeetCode 247 中心对称数2

回溯

回溯算法是一种试探算法，与暴力搜索的最大区别：在回溯算法中，是一步步向前试探，对于每一步的试探评估，在决定是否继续，可避免走弯路。

回溯算法的精华：出现非法的情况时，可退到之前的情景，可返回一步或多步；再去尝试别的路径和办法。

LeetCode 39.组合总和

动态规划

重要属性：（1）最优子结构：状态转移方程（2）重叠子问题

解决动态规划最难的两个地方：（1）如何定义 $f(n)$ ；（2）如何通过 $f(1)$ 、 $f(2)$... $f(n-1)$ 推导出 $f(n)$ 。

难点：（1）应当采用什么样的数据结构来保存什么样的计算结果。（思考：把问题的规模缩小，假设已经有人把所有子问题的结果得出最终结果）

动态规划的分类：

1.线性规划

子问题以线性分布，子问题的状态或结果可以保存在一维线性数据结构中；

通常会用 $dp[i]$ 表示第 i 个位置的结果或者从0开始到第 i 个位置为止的最佳状态或结果。

基本形式：1）当前所求的值仅仅依赖于有限个先前计算好的值，即 $dp[i]$ 依赖于 $dp[j], j < i$ 。比如斐波那契数列。 $dp[i] = dp[i-1] + dp[i-2]$;

LeetCode 70题，198题，62题。

2）当前所求的值仅仅依赖于所有先前计算和的值，即 $dp[i]$ 是各个 $dp[j]$ 的某种组合，其中 j 由0遍历到 $i-1$ 。

2.区间规划

1）各个子问题规模由不同区间来定义

2）子问题的最佳结果可以存储在二维数组中

LeetCode 516题

3.约束规划

0-1背包问题

二分搜索

输入数组或区间是有序的，且不会经常变动，要求从中找出一个满足条件的元素,这个时候就是要使用二分搜索。

核心要素：

1.确定范围和区间

2.取中间的数判断是否满足条件

3.如果不满足条件，判定应该往哪个半边继续进行搜索

LeetCode 33，34题

找模糊的边界问题：从数组 $\{-2, 0, 1, 4, 7, 9, 10\}$ 中找到第一个大于6的数。

贪婪算法

1.总是做出当前看来是最好的选择。

2.不从整体角度考虑问题，仅对局部的最优解，只有那些局部最优策略能产生全局最优策略的时候才能使用贪婪算法。

LeetCode 253 会议室2

LeetCode面试高频题目

全部为LeetCode，题号和题名

3.无重复字符的最长子串

4.寻找两个正序数组的中位数

215.数组中的第K个最大元素

23.合并K个排序链表

56.合并区间

435.无重叠区间

269.火星字典

772.基本计算器2

10.正则表达式匹配

84.柱状图中的最大矩形

28.实现strstr ()

336.回文对

340.至多包含 K 个不同字符的最长子串

407.接雨水 II