

plan

deepwaterooo

2021 年 10 月 29 日

目录

1	Weekly Plan	5
1.1	总结题型	5
1.2	比赛	6
1.3	数据规模与算法	6

Chapter 1

Weekly Plan

1.1 总结题型

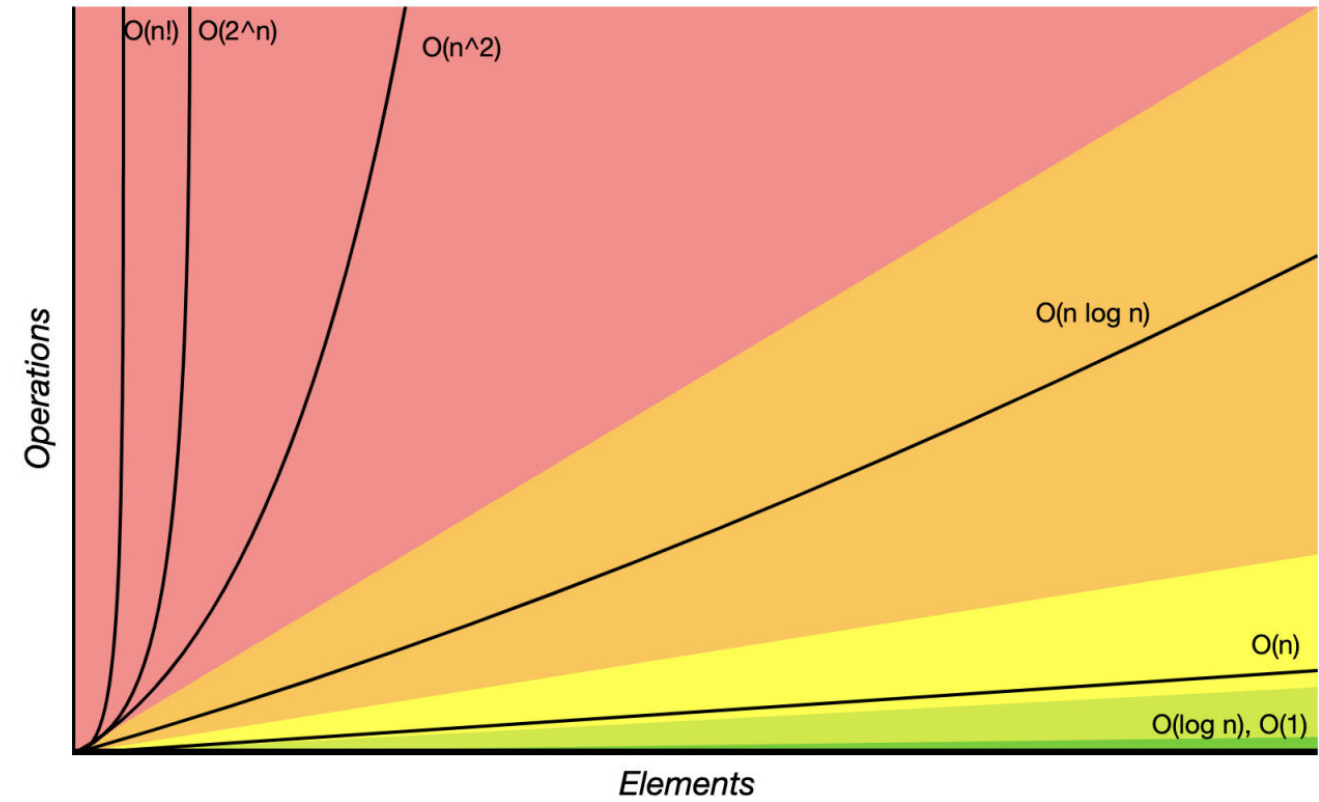
- 我可能有一点儿傻做题，而不太懂得总结与作笔记。所以在现在刷得已经差不多，我需要更多的时间花在巩固与总结题型，把做题的经历真正转化为解题的分析能力。所以接下来一两周可能会对重要题型作相关的分析与总结，希望能够显著提升我做题和比赛中的解题命中率。
- 这周刷得跟上一周差不多。计划接下来几周，每天刷 10 个题左右，一周 60-70 个题目，边刷边记错题进小本，到接下来的周六刷到 1150 左右；
- `dynamic programming`, `graph`, `segment tree` 等等剩下的题目都是高难度的精华，需要好好练习和消化
- 这个周主要复习了基础的部分，`hashing`, `hashtable` `string two pointers`, 以前很讨厌字符串，这个周写字符串看来，感觉基础的只要自己能够想得清楚的，都能很快实现出来，写得还比较得心应手。`TreeMap`, `TreeSet` 也用得比较顺一点儿了
- 写树的题目，`dfs`, `bfs graph` 各种 `node`, 现在也是写得很顺心了，只是通过不断地测试加强巩固
- 完成打基础的部分：`hashing`, `hashmap`, `string`, `two pointer`, `sliding window`, 这些基础部分的题目，希望扫完
- 如果某天头脑比较清醒、精力比较好的时候，会试图去慢速解决自己平时困难的地方：动态规划/`hashmap`/`hashing` 中数组的个数，不常用的算法等
- 数组相关的 `segment tree`, `binary index tree` 等的基础，希望能够理解得再彻底一些，到能活学活用的程度
- `Deque` 双端队列 $O(n)$ 解法的概念在建立，还需要很多的练习和熟悉
- 最讨厌扫描线，几个双数怎么也数不清楚 `heap` 等。。。。。这个周扫几个出去
- 如果说以前是迷迷糊糊刷题求 AC, 现在基本的概念在建立，希望从以前代码和题目的算法效率向代码优化中等偏优，寻求高效、最优解法的提升
- `bit manipulation`, `bitmasks` 基础知识基本掌握，还剩几道难题 `take my time` 慢慢解（感觉现在对 `bit` 操作，相对自信得心应手得多了！）
- `union find` 的几个题，基本算是基本扫完吧，剩下的几下慢慢写。。。。。
- 很喜欢现在自己搭建出来的 `window` 刷题环境：`WSL system`, `Zsh power shell`, `emacs configurations`, `locally everything`, except `Leetcode server is too slow`, `have to tolerant its latency`.....
- 至昨天晚上我终于意识到确定右侧单耳耳鸣，搬到现居住处后发病的（感觉现居住处到处都是电磁波干扰、洗衣烘衣的车床，厨房的冰箱，曾整小时整小时开过的洗手间风扇等），已经有几个周了。对于自 2013 年秋天野鸡大学的住宿环境以来，备受各种居住环境的困扰，尤其是 2019 年 9 月 10 月以来，我自小的听力受损，现在单侧耳鸣，可能的原因有家族遗传性高血压、遗传性脑血管肿瘤（外公舅舅和妈妈都受此脑溢血困扰过，大我五岁的亲姐姐前几个月也刚发此重病一次）等等。今天在网上稍搜索了一下相关信息，回想这几年的居住环境噪音和人为打扰与睡眠干扰、心里戚戚很不是滋味，希望我不至于会失去听力。
- But my suffering is still only my/a personal suffering. Unless I could find an appropriate job, nobody cares if you are sick or not. 因此，自信是本能，向往强大也是一种本能的向往。Anything happens, 我还是必须努力努力刷题，直到找到合适的工作。所以会近几周把剩下的一点儿题目刷完（`hard and medium only`, `keep easy untouched dont care`）不喜欢数字，也数字无缘，不打算写数组题目，如果一定要写，可能也只会把难题写一写吧

1.2 比赛

- 会尽量多参加一些比赛，比赛时的效率还是相对好一点儿，所以每周一次、每半月一次，以及以后 codeforces 上的比赛，希望都能够尽量地多参加一些。

1.3 数据规模与算法

Input Size	Complexity
50000	$O(n)$
20000	$O(n \log n)$
1000	$O(n^2)$
30	$O(n^4)$
16 (20)	$O(2^n)$



[illegible]

算法	时间复杂度			空间复杂度
	最佳	平均	最差	最差
<u>快速排序</u>	$O(n \log(n))$	$O(n \log(n))$	$O(n^2)$	$O(\log(n))$
<u>归并排序</u>	$O(n \log(n))$	$O(n \log(n))$	$O(n \log(n))$	$O(n)$
<u>Timsort</u>	$O(n)$	$O(n \log(n))$	$O(n \log(n))$	$O(n)$
<u>堆排序</u>	$O(n \log(n))$	$O(n \log(n))$	$O(n \log(n))$	$O(1)$
<u>冒泡排序</u>	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$
<u>插入排序</u>	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$
<u>选择排序</u>	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(1)$
<u>希尔排序</u>	$O(n)$	$O((n \log(n))^2)$	$O((n \log(n))^2)$	$O(1)$
<u>桶排序</u>	$O(n + k)$	$O(n + k)$	$O(n^2)$	$O(n)$
<u>基数排序</u>	$O(nk)$	$O(nk)$	$O(nk)$	$O(n + k)$

节点 / 边界管理	存储	增加顶点	增加边界	移除顶点	移除边界	查询
<u>邻接表</u>	$O(V + E)$	$O(1)$	$O(1)$	$O(V + E)$	$O(E)$	$O(V)$
<u>邻接矩阵</u>	$O(V ^2)$	$O(V ^2)$	$O(1)$	$O(V ^2)$	$O(1)$	$O(1)$

类型	时间复杂度						
	建堆	查找最大值	分离最大值	提升键	插入	删除	合并
<u>(排好序的) 链表</u>	–	$O(1)$	$O(1)$	$O(n)$	$O(n)$	$O(1)$	$O(m + n)$
<u>(未排序的) 链表</u>	–	$O(n)$	$O(n)$	$O(1)$	$O(1)$	$O(1)$	$O(1)$
<u>二叉堆</u>	$O(n)$	$O(1)$	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$	$O(m + n)$
<u>二项堆</u>	–	$O(1)$	$O(\log(n))$	$O(\log(n))$	$O(1)$	$O(\log(n))$	$O(\log(n))$
<u>斐波那契堆</u>	–	$O(1)$	$O(\log(n))$	$O(1)$	$O(1)$	$O(\log(n))$	$O(1)$