

2025 年秋季学期算法基础期中考试

学号 _____ 姓名 _____

主定理： 令 $a \geq 1$ 和 $b > 1$ 是常数， $f(n)$ 是一个函数， $T(n)$ 是定义在非负整数上的递归式：

$$T(n) = aT(n/b) + f(n)$$

其中我们将 n/b 解释为 $\lfloor n/b \rfloor$ 或 $\lceil n/b \rceil$ 。那么 $T(n)$ 有如下渐进界：

1. 若对某个常数 $\epsilon > 0$ 有 $f(n) = O(n^{\log_b a - \epsilon})$ ，则 $T(n) = \Theta(n^{\log_b a})$ 。
2. 若对整数 $k \geq 0$ 有 $f(n) = \Theta(n^{\log_b a} \lg^k n)$ ，则 $T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$ 。
3. 若对某个常数 $\epsilon > 0$ 有 $f(n) = \Omega(n^{\log_b a + \epsilon})$ ，且对某个常数 $c < 1$ 和所有足够大的 n 有 $af(n/b) \leq cf(n)$ ，则 $T(n) = \Theta(f(n))$ 。

一、**判断题**（根据表述判断正误，并简要说明理由；每题 5 分，共 30 分；每一题判断答案正确得 2 分，解释理由正确得 3 分。）。

1. (T, F) 递归式 $T(n) = 2T(\frac{2n}{3}) + n \log n$ 的解满足 $T(n) = \Theta(n^2)$ 。
2. (T, F) 记 $\sigma(n)$ 表示数字 n 的因子个数，例如 $\sigma(6) = 4$ ，因为 1, 2, 3, 6 都是 6 的因子。那么 $\sum_{i=1}^n \sigma(i)$ 是 $\Theta(n\sqrt{n})$ 的。
3. (T, F) 归并排序是稳定的，其时间复杂度在任何情况下都一定是 $\Theta(n \log n)$ 。

4. (T, F) 计数排序 (Counting Sort) 的空间复杂度是 $O(1)$ 。
5. (T, F) 某二叉树的前序、中序、后序遍历中, 任意给定两项, 都可以唯一确定第三项。
6. (T, F) 对于一棵黑高为 h 的红黑树, 其内部节点 (也就是有键值的点) 的数量至多是 $2^{h+1} - 1$ 。

二、简答题 (根据题目要求写出解答过程; 每题 10 分, 共 40 分)。

1. 对于递归式 $T(n) = 2T(\frac{n}{2}) + \frac{n}{\log_2 n}$, 请给出渐近确界。

2. 有一排共 n 桶染料，每桶染料的颜色用一个 $[1, n]$ 之间的数字表示。画家小 S 想要使用至少 k 种颜色作画，但是他只能使用**连续的**若干桶染料（也就是若干桶相邻的染料）。现在请你求出：他最少需要用多少桶染料，才能够满足至少 k 种颜色的要求。给出伪代码，并说明算法的工作原理。你的算法时间复杂度需要是 $O(n)$ 的，超出会酌情扣分。

注意，你的算法不需要考虑无解的情况。

3. 已知在一个区间集合中，被最多数量区间覆盖的点被称为最大重叠点，且它一定可以是某个区间的端点。请你简述一个数据结构的设计思路，使它能够支持动态查询一个动态区间集合中的最大重叠点（任意一个符合要求的端点即可），并分析各操作的时间复杂度。

提示：可将左端点关联 +1 值，右端点关联-1 值。

4. 对于图一所给的斐波那契堆, 黑色节点表示结点的 mark 属性为 true:
- 1) 抽取最小结点, 画出操作后最后的结果 (需要考虑 CONSOLIDATE(合并) 操作)。
 - 2) 在 1) 的基础上, 把关键字为 30 的节点的值降低为 4, 画出操作后的最后的结果 (需考虑 CASCADING-CUT(级联切断) 操作)。

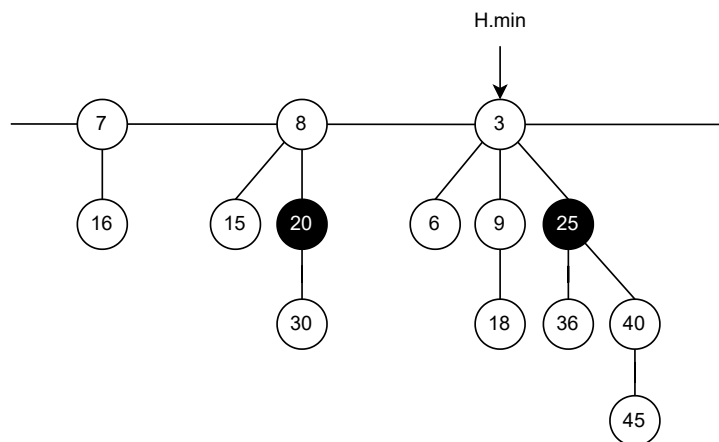


图 1: 斐波那契堆

三、综合题（根据题目要求写出解答过程；每题 15 分，共 30 分）。

1. 线性排序算法

(a) 基数排序 (Radix Sort) 的正确性依赖于一个稳定的子排序算法。假设我们使用计数排序 (Counting Sort) 作为 RADIX-SORT 所需的稳定子排序算法 STABLE-SORT，请写出 COUNTING-SORT(A, d) 的伪代码，该算法用于对一个包含 n 个元素的数组 A 进行排序，其中每个元素都是不大于 m 的非负整数。

Algorithm 1 COUNTING-SORT(A, m)

1:
2:
3:
4:
5:
6:
7:
8:
9:
10:
11:
12:
13:
14:

(b) 已知待排序的 n 个元素中的每一个都有 d 位数字，且每一位数字都可以取 0 到 $k-1$ 中的任意一个值（即最多有 k 个可能的值），请推导出 RADIX-SORT 的总时间复杂度。

(c) 在实际应用中，我们经常处理 b 位的二进制数。一种高效的策略是不按 1 位排序，而是将 b 位分成 d 组，每组 r 位（其中 $r \leq b$ ）。我们对这 $d = \lceil b/r \rceil$ 组分别使用基数排序（同样使用计数排序作为子算法）。此时，RADIX-SORT 的总时间复杂度 $T(n, b, r)$ 是多少？

2. 在标准二叉搜索树中, 我们可以高效支持 SEARCH、INSERT 和 DELETE 等操作, 但无法高效处理顺序统计问题, 如 “第 i 小的关键字是什么?” 或者 “关键字 x 的排名是多少?”

为解决此问题, 我们引入顺序统计树 (Order-Statistic Tree, OST): 它在每个结点 x 中额外维护一个属性 $x.size$, 表示以 x 为根的子树中包含的结点总数 (包括 x 自身)。空指针 (记为 NIL) 的 $size$ 定义为 0。利用 $size$ 信息可实现 OS-SELECT(x, i), 返回以 x 为根的子树中第 i 小的关键字对应的结点 ($1 \leq i \leq x.size$)。

(a) 在一棵 OST 中, 某结点 x 满足 $x.left.size = 3$, $x.right.size = 5$ 。

1. 计算 $x.size$ 。

2. 若调用 OS-SELECT($x, 4$), 将返回哪个结点? 简要说明理由。

(b) 用伪代码实现函数 OS-SELECT(x, i), 其中 x 是一棵 OST 的根结点, i 为正整数。若 $1 \leq i \leq x.size$, 返回对应结点; 否则返回 NIL。

(c) 假设 OST 由关键字 $1, 2, \dots, n$ 按均匀随机排列插入构建。令 D_i 表示第 i 小关键字对应结点的深度 (根深度为 0)。

1. 证明: 对任意 $1 \leq i \leq n$,

$$\mathbb{E}[D_i] = H_i + H_{n-i+1} - 2,$$

(提示: $\ln(k+1) \leq H_k \leq 1 + \ln k$ 对所有整数 $k \geq 1$ 成立, 其中 $H_k = \sum_{j=1}^k \frac{1}{j}$ 为第 k 个调和数。)

2. 利用 (1) 的结论, 推导 OS-SELECT($T.root, i$) 的时间复杂度。

四、附加题（根据题目要求写出解答过程，共 10 分）。

你得到了 n 块按顺序排成一排的「记忆碎片」，第 i 块碎片上记录的记忆发生时间为 t_i 。已知 $t_1, t_2, \dots, t_n$ 是 $1 \sim n$ 的一个排列，也就是说，这 n 块碎片本来应该是按 $1, 2, \dots, n$ 的时间顺序出现的，但现在被打乱了。你希望通过若干次交换操作，把这些碎片恢复成严格单调递增的顺序：

$$(t_1, t_2, \dots, t_n) \longrightarrow (1, 2, \dots, n).$$

你拥有一个「回忆能力值」 k 。一次操作中，你可以任选两块碎片，设它们记忆的时间分别是 x 和 y ，当且仅当 $|x - y| \leq k$ 时，你可以交换它们现在所在的位置。交换的体力消耗规则如下：

- 若 $|x - y| = 1$ ，也就是这两段记忆发生在相邻的两个时间点上，由于发生时间过于靠近的两块碎片很难回想，这次交换需要消耗你 **1 点**精力；
- 若 $2 \leq |x - y| \leq k$ ，则这次交换**不需要**消耗精力，代价为 0。

你可以进行任意多次满足条件的交换操作。请你计算：为了将碎片恢复成 $(1, 2, \dots, n)$ ，你最少需要消耗多少点精力。

设计时间复杂度尽可能小的算法。（输入： $n, [t_1, t_2, \dots, t_n]$ ， k ；输出：最少需要消耗的精力值）