2023 年秋季学期算法基础期中考试（样卷）

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 $\varepsilon > 0$ 有 $f(n) = O(n^{\log_b a - \varepsilon})$，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 $\varepsilon > 0$ 有 $f(n) = \Omega(n^{\log_b a + \varepsilon})$，且对某个常数 $c < 1$ 和所有足够大的 n 有 $af(n/b) \leq cf(n)$，则 $T(n) = \Theta(f(n))$。

Master Theorem: Let $a \geq 1$ and $b > 1$ be constants and $f(n)$ be a function. Let $T(n)$ be defined on the nonnegative integers by the following recurrence

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

Notice that here n/b can be interpreted as either $\lfloor n/b \rfloor$ or $\lceil n/b \rceil$. Then $T(n)$ can be bounded asymptotically as follows:

1. If there exists a constant $\varepsilon > 0$ such that $f(n) = O(n^{\log_b a - \varepsilon})$ then $T(n) = \Theta(n^{\log_b a})$.
2. If there exists an integer $k \geq 0$ such that $f(n) = \Theta(n^{\log_b a} \lg^k n)$ then $T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$.
3. If there exists a constant $\varepsilon > 0$ such that $f(n) = \Omega(n^{\log_b a + \varepsilon})$, and if $af(n/b) \leq cf(n)$ for some constant $c < 1$, then $T(n) = \Theta(f(n))$.

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）。

1. 递归式 $T(n) = 7T(\frac{n}{2}) + n^2$ 的解为 $T(n) = \Theta(n^2)$。 (　　)

The solution of the recurrence $T(n) = 7T(\frac{n}{2}) + n^2$ is $T(n) = \Theta(n^2)$.

2. 给定一个包含 n 个整数的数组 A，归并排序总是可以在最坏情况下用 $O(n \lg n)$ 的时间对数组 A 进行排序。 ( )
Given an array A of n integers, merge sort can always sort the array A in time $O(n \lg n)$ in the worst case.

3. 假设有一个包含 n 个待排序数据记录的数组，且每条记录的关键字的值为 0 或 1。对这样一组记录进行排序，存在时间代价为 $O(n)$，稳定的原址（除了输入数组外，算法只需要固定的额外存储空间）排序算法。 ( )
Suppose that we have an array of n data records to sort and that the key of each record has the value 0 or 1. There exists such an algorithm for sorting such a set of records that satisfies the following three characteristics: 1) The algorithm runs in $O(n)$ time. 2) The algorithm is stable. 3) The algorithm sorts in place, using no more than a constant amount of extra storage space.

4. 从一个由 n 个互异元素构成的数组中选择第 i 个 $(i > 1)$ 顺序统计量和找最小值的渐近运行时间相同。 ( )
Given an array A of n distinct elements, the asymptotic running time for selecting the ith element and selecting a minimum is the same.

5. T_1, T_2 是相同集合上的两棵二叉搜索树，若 T_1, T_2 的前序遍历序列相同，则两棵树相同。 ( )
Given two binary search trees T_1 and T_2 on the same set. If the inorder traversals of T_1 and T_2 are the same, they are the same tree.

6. AVL 树是一种高度平衡的二叉搜索树：对于每一个结点 x, x 的左子树和右子树的高度差至多为 1。红黑树也是 AVL 树。 ( )
An AVL tree is a binary search tree that is height balanced: for each node x, the heights of the left and right subtrees of x differ by at most 1. The red-black tree also is an AVL tree.

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

1. 我们在求解算法的时间复杂度时，通常假设：**过程调用中的参数传递**花费常量的时间，即使传递一个 N 个元素的数组也是如此。考虑这样一种参数传递策略：传递数组时，只复制过程可能访问的子区域，若数组 $A[p..q]$ 被传递，则时间为 $\Theta(q-p+1)$。请给出**归并排序**在该种参数传递策略下的**最坏情况**运行时间的递归式。
We assume that parameter passing during procedure calls takes constant time, even if an $N-$element array is being passed. We consider such a parameter-passing strategy: An array is passed by copying only the subrange that might be accessed by the called procedure. Time $=\Theta(q-p+1)$ if the subarray $A[p..q]$ is passed. Please give recurrences for the worst-case running times of merge-sort when arrays are passed using aforementioned method.

2. 给定两个 n 位整数 X 和 Y。命题：计算 XY，我们需要 $\Omega(n^2)$ 次的一位整数的加法和乘法。请问该命题是否正确？如果不正确请给出你的答案。
Given two integers X, Y, each of n digits. Proposition: to compute the production XY, we always need to use $\Omega(n^2)$ additions and multiplications of one-digit integers. Is this proposition correct? If not, please give your answer.

3. 对一个包含 n 个元素的集合，k 分位数是指能把有序集合分成 k 个等大小集合的第 $k-1$ 个顺序统计量。给出一个能找出某一集合的 k 分位数的 $O(n\lg k)$ 时间的算法。
The kth quantiles of an n-element set are the $k-1$ order statistics that divide the sorted set into k equal-sized sets (to within 1). Give an $O(n\lg k)$-time algorithm to find the kth quantiles of a set.

4. 对于图 1 所示的斐波那契堆，给出执行抽取最小结点操作之后的结果。
Please give the result of extracting the minimum node of the Fibonacci heap shown in Fig. 1.

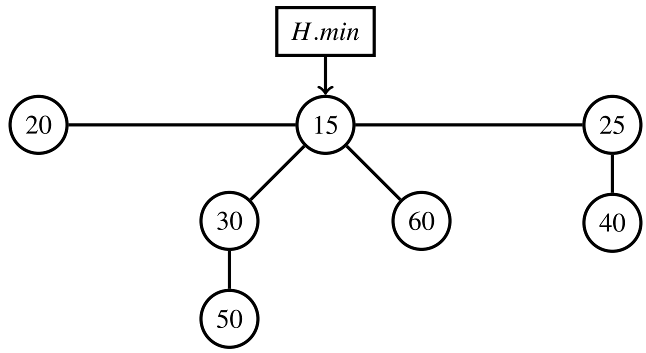

图 1: 斐波那契堆
Fig. 1 Fibonacci Heap

三、**综合题**（根据题目要求写出解答过程；每题 15 分，共 30 分）。

1. 排序和顺序统计量的计算在数据分析领域有着十分重要的作用，请回答下列问题：

(a) 给定一个包含 n 个互异的元素的集合，请简要描述如何在期望时间为 $\Theta(n)$ 的时间内找到第 k 小的元素。

(b) 设计一个算法，按顺序输出前 k 个最小的元素，简要描述该算法的思想并给出时间复杂度。要求该算法的时间复杂度小于 $\Theta(kn)$。

(c) 给定两个分别包含 n 个不同元素的有序序列 X 和 Y，请设计一个 $\Theta(\lg n)$ 时间的算法，找到 X, Y 序列中所有元素的中位数。

1. Sorting and the calculation of ordinal statistics play an important role in data analysis. Please answer the following questions.

(a) Given a set of n distinct elelments, please discribe briefly that how to find ith small elements within expected running time $\Theta(n)$-time.

(b) Find the algorithm to find the ith smallest in sorted order. Describe the algorithm briefly and give the running time complexity. The time complexity of this algorithm should be less than $\Theta(kn)$.

(c) Let X, Y be two arrays, each containing n distinct numbers already in sorted order. Give an $O(\lg n)$-time algorithm to find the median of all $2n$ elements in arrays X and Y.

2. 一棵二叉搜索树 T 的前序遍历如下所示：

$$56, 26, 18, 12, 24, 30, 27, 28, 29, 200, 190, 213$$

(a) 画出一棵满足该前序遍历的二叉树 T。

(b) 是否可以给这棵树的结点着色使其成为一棵红黑树 T_1。若可以，画出 T_1，若不可以，说明理由。

(c) 将结点 30 从 T 中删除得到 T_2，画出 T_2。

Assume that there is a binary search tree T of 12 elements. After we do pre-order tree traversal of T and print the elements in the order of visiting, we have the following output.

$$56, 26, 18, 12, 24, 30, 27, 28, 29, 200, 190, 213$$

(a) Draw an original binary search tree T that will produce such output.

(b) Can the nodes of T be colored to be a red-black tree T_1. Draw T_1 or argue why it can't.

(c) Draw the binary search tree T_2 after we delete the number 30 from T.

四、附加题（根据题目要求写出解答过程；每题 10 分，共 10 分）。

定义 Josephus 问题如下：假设 n 个人围成一个圆圈，给定一个正整数 m 且 $m \leq n$。从某个指定的人开始，沿环将遇到的每第 m 个人移出队伍，每个人移出之后，继续沿环数剩下来的人。这个过程直到所有的 n 个人都被移出后结束。每个人移出的次序定义了一个来自整数 $1,2,...,n$ 的 $(n,m)-$Josephus 排列。例如，$(7,3)-$Josephus 排列为 $(3,6,2,7,5,1,4)$。

(a) 假设 m 为一个常数，描述一个时间复杂度 $O(n)$ 的算法，使得对于给定的 n，能够输出 $(n,m)-$Josephus 排列。

(b) 假设 m 不是常数，简要描述时间复杂度为 $O(n\lg n)$ 的算法，使得对于给定的 n，能够输出 $(n,m)-$Josephus 排列。

We define the Josephus problem as follows. Suppose that n people form a circle and that we are given a positive integer $m \leq n$. Beginning with a designated first person, we proceed around the circle, removing every mth person. After each person is removed, counting continues around the circle that remains. This process continues until we have removed all n people. The order in which the people are removed from the circle defines the (n,m)-Josephus permutation of the integers$1,2,...,n$. For example, the$(7,3)$-Josephus permutation is $(3,6,2,7,5,1)$.

(a) Suppose that m is a constant. Describe an $O(n)$-time algorithm that, given an integer n, outputs the (n,m)-Josephus permutation.

(b) Suppose that m is not a constant. Describe an $O(n\lg n)$-time algorithm that, given integers n and m, outputs the (n,m)-Josephus permutation.