算法导论习题选集

作业 8

节选自《算法导论》教材第三版

课程网站：https://algorithm.cuijiacai.com

Problem 1

（有序序列中的 i 个最大数）给定一个包含 n 个元素的集合，我们希望利用基于比较的算法找出按顺序排列的前 i 个最大元素。请设计能实现下列每一项要求，并且具有最佳渐近最坏情况运行时间的算法，以 n 和 i 来表示算法的运行时间：

1. 对输入数据排序，并找出前 i 个最大数；

2. 对输入数据建立一个最大优先队列，并调用 EXTRACT-MAX 过程 i 次。

3. 利用一个顺序统计量算法来找到第 i 大的元素，然后用它作为主元划分输入数组，再对前 i 大的数排序。

Problem 1 continued on next page…

(续页)

Problem 2

（带权中位数）对分别具有正权重 $w_1, w_2, \cdots, w_n$ ，且满足 $\sum_{i=1}^{n} w_i = 1$ 的 n 个互异元素 $x_1, x_2, \cdots, x_n$ 来说，带权中位数 x_k （较小中位数）是满足如下条件的元素：

$$\sum_{x_i < x_k} w_i < \frac{1}{2}$$

和

$$\sum_{x_i > x_k} w_i \le \frac{1}{2}$$

例如，如果元素是 $0.1, 0.35, 0.05, 0.1, 0.15, 0.05, 0.2$ ，并且每个元素的权重等于本身（即对所有 $i = 1, 2, \cdots, 7$ ，都有 $w_i = x_i$），那么中位数是 0.1 ，而带权中位数是 0.2 。

1. 证明：如果对所有的 $i = 1, 2, \cdots, n$ ，都有 $w_i = 1/n$ ，那么 $x_1, x_2, \cdots, x_n$ 的中位数就是 x_i 的带权中位数。

2. 利用排序，设计一个最坏情况下 $O(n \log n)$ 时间的算法，可以得到 n 个元素的带权中位数。

3. 说明如何利用像第 8 讲 PPT 第 14 页的 SELECT 这样的线性时间中位数算法，在 $\Theta(n)$ 最坏情况时间内求出带权中位数。

邮局位置问题的定义如下：给定权重分别为 $w_1, w_2, \cdots, w_n$ 的 n 个点 $p_1, p_2, \cdots, p_n$ ，我们希望找到一个点 p （不一定是输入点中的一个），使得 $\sum_{i=1}^{n} w_i d(p, p_i)$ 最小，这里 $d(a, b)$ 表示点 a 与 b 之间的距离。

4. 证明：对一维邮局位置问题，带权中位数是最好的解决方法，其中，每个点都是一个实数，点 a 与 b 之间的距离是 $d(a, b) = |a - b|$ 。

5. 请给出二维邮局位置问题问题的最好解决方法：其中的点是 (x, y) 的二维坐标形式，点 $a = (x_1, y_1)$ 与 $b = (x_2, y_2)$ 之间的距离是 **Manhattan 距离**，即 $d(a, b) = |x_1 - x_2| + |y_1 - y_2|$ 。

Problem 2 continued on next page…

(续页)

Problem 2 continued on next page…

(续页)

Problem 3

（小顺序统计量）要在 n 个数中选出第 i 个顺序统计量，SELECT（详见第 8 讲 PPT 第 14 页）在最坏情况下需要的比较次数 $T(n)$ 满足 $T(n) = \Theta(n)$ 。但是，隐含在 Θ 记号中的常数项是非常大的。当 i 相对 n 来说很小时，我们可以实现一个不同的算法，它以 SELECT 作为子程序，但在最坏情况下所做的比较次数要更少。

1. 设计一个能用 $U_i(n)$ 次比较在 n 个元素中找出第 i 小元素的算法。其中，

$$U_i(n) = \begin{cases} T(n) & \text{if } i \geq n/2 \\ \\ \lfloor n/2 \rfloor + U_i(\lceil n/2 \rceil) + T(2i) & \text{otherwise} \end{cases}$$

（提示：从 $\lfloor n/2 \rfloor$ 个不相交对的两两比较开始，然后对由每对中的较小元素构成的集合进行递归。）

2. 证明：如果 $i < n/2$ ，则有 $U_i(n) = n + O(T(2i)\log(n/i))$ 。

3. 证明：如果 i 是小于 $n/2$ 的常数，则有 $U_i(n) = n + O(\log n)$ 。

4. 证明：如果对所有 $k \geq 2$ 有 $i = n/k$ ，则 $U_i(n) = n + O(T(2n/k)\log k)$ 。

Problem 3 continued on next page…

(续页)

Problem 4

（随机化选择的另一种分析方法）在这个问题中，我们用指示器随机变量来分析 RANDOMIZED-SELECT，这一方法类似于第 6 讲 PPT 第 27 页中所用的对 RANDOMIZED-QUICKSORT 的分析方法。

与快速排序中的分析一样，我们假设所有的元素都是互异的，输入数组 A 的元素被重命名为 $z_1, z_2, \cdots, z_n$，其中 z_i 是第 i 小的元素。因此，调用 RANDOMIZED-SELECT$(A, 1, n, k)$ 返回 z_k。

对所有 $1 \leq i < j \leq n$，设

$$X_{ijk} = I\{\text{在执行算法查找} z_k \text{ 期间，} z_i \text{ 与} z_j \text{ 进行过比较}\}$$

1. 给出 $E[X_{ijk}]$ 的准确表达式。（提示：你的表达式可能有不同的值，依赖于 i, j, k 的值。）

2. 设 X_k 表示在找到 z_k 时 A 中元素的总比较次数，证明：

$$E[X_k] \leq 2\left(\sum_{i=1}^{k}\sum_{j=k}^{n}\frac{1}{j-i+1} + \sum_{j=k+1}^{n}\frac{j-k-1}{j-k+1} + \sum_{i=1}^{k-2}\frac{k-i-1}{k-i+1}\right)$$

3. 证明：$E[X_k] \leq 4n$。

4. 假设 A 中的元素都是互异的，证明：RANDOMIZED-SELECT 的期望运行时间是 $O(n)$。

Problem 4 continued on next page…

(续页)