算法设计与分析第三章学习指南

视频

https://www.icourse163.org/course/HIT-356006

算法设计与分析(基础篇) 第三讲

阅读

算法导论(第三版) 4.1 节，4.2 节，第 9 章，第 30 章，33.3 节，33.4 节，第 27 章 (选学)

练习题

3.1 给定平面上 n 个点，求其中的一点对，使得在 n 个点的所有点对中，该点对的距离最小。严格地说，最接近点对可能多于 1 对。为了简单起见，这里只限于找其中的一对。

（1）设计一个时间复杂度为 $O(n^2)$的算法求距离最近的点对，要求写出算法伪代码；

（2）利用分治的思想设计一个时间复杂度为 $O(n\log n)$的算法求距离最近的点对，要求写出算法伪代码。

3.2 对于平面上的两个点 p1=(x1, y1)和 p2=(x2,y2)，如果 x1<=x2 且 y1<=y2，则 p2 支配 p1，给定平面上的 n 个点，请设计算法求其中没有被任何其他点支配的点。

3.3 设计一个分治算法，在一个 2 维平面上求 n 个点中距离最近的两个点，要求时间复杂性是 $o(n^2)$，请写出算法伪代码并分析时间复杂性。

3.4 阅读 https://oi-wiki.org/math/quick-pow/学习基于分治思想的“快速幂”算法。设计一个通过矩阵运算，在 O(logN)时间内计算斐波那契数列第 N 项的算法。

3.5 给定一个数组 A，任务是设计一个算法求得数组中的“主元素”，即在数组中个数超过数组总元素个数一半的元素。但是数组中元素的数据类型可能是复杂类型，这意味着数组中的元素进能够比较是否相等而不存在序关系，设对于两个元素 A[i]和 A[j]，判定是否 A[i]=A[j]需要常数时间。

(1) 设计一个时间复杂性为 O(n log n)的算法解此问题

(2) 设计一个时间复杂性为 O(n)的算法解此问题.

3.6 对于给定的 n 个元素的数组 A[1..n],要求从中找出第 k 小的元素。请设计分治算法解决这个问题，要求算法的平均时间复杂度是 O(n)。答案要求包含以下内容：（1）用简明的自然语言表述算法的基本思想；（2）用伪代码描述算法；（3）分析算法的时间复杂度。

3.7 给定实数数组 $A[1:n]$，试设计一个分治算法找出其中的最小元素和最大元素，使得比较操作的总次数严格小于 $2(n-1)$。答案要求包含以下内容：（1）用简明的自然语言表述算法的基本思想；（2）用伪代码描述算法；（3）分析算法的时间复杂度。

3.8 有长度为 N 的数组 A、B，每个数组中存储的浮点数已经升序排列。设计一个 O(logn)时间的分治算法，找出这 2n 个数的中位数。证明算法的正确性。

3.9 设计分治算法求解下述问题：

输入：一个一维整数数组 A(其中元素可能是正数也可能是负数)

输出：A 的连续子数组，要求其和最大

例如，输入数组是{-2, -5, 6, -2, -3, 1, 5, -6}，其结果是{6, -2, -3, 1, 5},其和为 7.

要求: (1) 算法时间复杂度为 O(nlogn)；(2) 写出算法伪代码；(3) 分析算法时间复杂度

3.10 设计一个“三路归并”的排序算法，并分析它的时间复杂性。

3.11 Hadamard 矩阵 $H_0, H_1, H_2\cdots$递归定义如下:

H_0是1×1的矩阵[1];

对于 k>0, H_k是$2^k\times2^k$的矩阵$\begin{bmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & -H_{k-1} \end{bmatrix}$,

设 v 是一个长度为 2^k的列向量，设计一个时间复杂性为 O(nlogn)的算法计算矩阵-向量乘法 $H_k v$(设单个数字的加法和乘法都在单位时间内完成)。

3.12 已知在一个 $2^k * 2^k$（k>0）个方格组成的棋盘中，若恰有一个方格与其它方格不同，则称该方格为一特殊方格，称该棋盘为一特殊棋盘。图 1 所示的特殊棋盘为 k=2 时的一个特殊棋盘。现在要用图 2 中 4 种不同形态的 L 型骨牌覆盖一个给定的特殊棋盘上除特殊方格以外的所有方格，且任何 2 个 L 型骨牌不得重叠覆盖。

（1）利用分治的思想设计一个算法解决棋盘覆盖问题；

（2）分析该算法的时间复杂度。

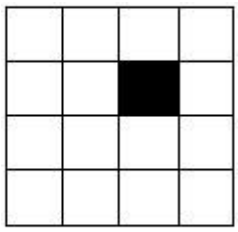


图 1 k=2 的一个特殊棋盘

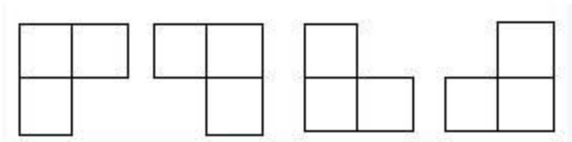
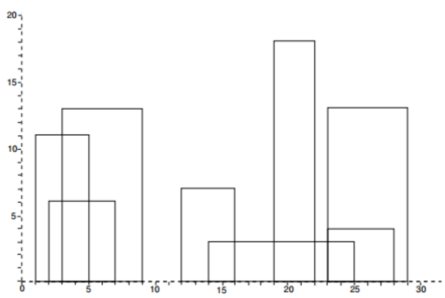


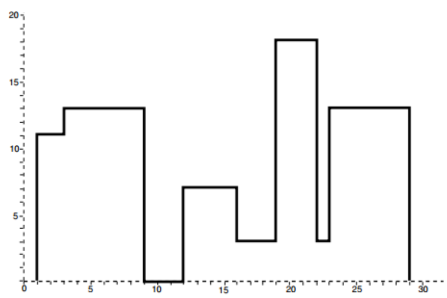
图 2 四种 L 型骨牌

3.13. 假设建筑表示为三元组(x_L, H, x_R)，其中 x_L,x_R 表示建筑左、右两侧的 *x* 坐标,*H* 表示建筑的高度。如下左侧图中的建筑可以表示为三元组序列{(**3**, 13, **9**),(**1**, 11, **5**),(**12**, 7, **16**),(**14**, 3, **25**),(**19**, 18, **22**),(**2**, 6, **7**),(**23**, 13, **29**),(**23**, 4, **28**)。建筑群的 skyline 是一系列横坐标以及连接相邻横坐标的水平线高度来表示，下图右侧的 skyline 表示为整数序列(**1**, 11, **3**, 13, **9**, 0, **12**, 7, **16**, 3, **19**, 18, **22**, 3, **23**, 13, **29**, 0)。考虑设计分治算法求解如下计算问题。给出分治算法的伪代码描述，并分析算法的时间复杂度。

问题定义：输入：*n* 幢建筑构成的建筑群；**输出：***n* 幢建筑的 skyline;



buildings



skyline

3.14 逆序对数求解：有长度为 N 的浮点数组 A，元素分别为 $a_1, a_2, ..., a_N$。如果满足 i<j 且 $a_i>a_j$，则(a_i, a_j)构成一个逆序对。设计分治方法求解数组 A 的逆序对个数，并分析算法的时间复杂

性。

3.15 给定一棵有 N 个节点的树，树上的每条边都有权值。定义两个节点 v_i, v_j 间的距离 dis(v_i, v_j)为节点间路径的权值和。设计分治方法求解：树上有多少个节点对(v_i, v_j)满足 i<j、且 dis(v_i, v_j)<=K。

3.16 线段树（segment tree）是用来存放给定区间（segment, or interval）内对应信息的一种数据结构。树上的每个节点代表一个区间，节点的左右子树二分地将区间分为两部分。在一棵线段树上，可以在 O(logN)时间内完成“单点数据修改”、“区间最值查询”、“区间求和查询”等操作。通过阅读相关资料，理解线段树的基本工作原理后，请尝试设计一种线段树上的“区间修改算法”，使其可以在 O(logn)时间内完成“对一个区间内每个数都增加一固定值”的操作，并仍能使“区间最值查询”、“区间求和查询”操作能够在 O(logN)时间内完成。