动态规划 练习题

1. 考虑三个字符串 X,Y,Z 的最长公共子序列 LCS(X,Y,Z)。

(1) 寻找反例 X,Y,Z 使得 LCS(X,Y,Z)，LCS(X, LCS(Y,Z));

(2) 设计动态规划算法计算X,Y,Z的最长公共子序列,分析算法的时间复杂度。

1. 输入数组 A[0:n] 和正实数d, 试设计一个动态规划算法输出 A[0:n]的一个最长子序列, 使得子序列中相继元素之差的绝对值不超过 d。分析算法的时间复杂度。
2. 给定一个整数序列 a1 ,...,an 。相邻两个整数可以合并, 合并两个整数的代价是这两个整数之和。通过不断合并最终可以将整个序列合并成一个整数, 整个过程的总代价是每次合并操作代价之和。试设计一个动态规划算法给出a1 ,...,an 的一个合并方案使得该方案的总代价最大。答案应包括:

(1) 用简明的语言表述这个问题的优化子结构;

(2) 根据优化子结构写出代价方程;

(3) 根据代价方程写出动态规划算法 (伪代码) 并分析算法的时间复杂性;

1. 输入表达式 a1O1 a2 O2 ···On−1 an, 其中 ai 是整数 (1≤i≤n), Oj∈{+,−,×} (1≤i≤ n−1)。设计一个动态规划算法, 输出一个带括号的表达式 (不改变操作数和操 作符的次序), 使得表达式的值达到最大, 分析算法的时间复杂性。
2. 设平面上有一个 m × n 的网格, 将左下角的网格点标记为 (0,0)，而右上角的网格点标记为 (m,n)。某人想从 (0,0) 出发沿网格线行进到达 (m,n),但是在网格点 (i,j) 处他只能向上行进或者向右行进, 向上行进的代价为aij (amj = +∞), 向右行进的代价是 bij (bin = +∞)。试设计一个动态规划算法, 在这个网格中为该旅行者寻找一条代价最小的旅行路线。
3. 给定一个 n × n 的矩阵 A, 矩阵中的元素只取 0 或者 1。设计一个动态规划算法, 求解得到 A 中元素全是 1 的子方阵使其阶数达到最大值。
4. 输入平面上 n 个点, 点与点之间的距离定义为欧几里得距离。试设计一个动态规划算法输出一条先从左到右再从右到左的一条最短路径, 使得每个输入点恰好被访问一次。
5. 输入一棵加权树 T, 其中每条边的权值均是实数, 试设计一个动态规划算法输出权值最大的子树.