2021年算法分析与设计第1次课后作业

Due: Oct 28, 2021

1. 求解下列递归式的渐进解；

(1)  (2) 

(3) 

解：(1) 利用主方法求解

所以

(2) 利用直接展开法求解

因为

所以

(3) 利用主方法求解

所以

2. 将下列6个函数按渐近增长率由低至高进行排序，要求写出判断依据

    （各log函数的底数都为2）

解：

因此，从小到大排序为

3. 数组a[*n* + 1]的每个下标作为一个阶梯，第*i*个阶梯对应着一个非负数的体力花费值cost[*i*]（数组下标从0开始，即0 ≤ *i* ≤ *n*）。每当你爬上一个阶梯你都要花费对应的体力值，一旦支付了相应的体力值，你就可以选择向上爬一个阶梯或者爬两个阶梯。请你设计算法求解达到楼层顶部（即第*n*个阶梯）的最低花费。在开始时，你可以选择从下标为0或1的元素作为初始阶梯。要求给出子问题定义和递归关系式。

解：

假设数组cost 的长度为*n*，则*n*个阶梯分别对应下标0到 *n* – 1，楼层顶部对应下标*n*，问题等价于计算达到下标*n*的最小花费。可以通过动态规划求解。

创建长度为*n* + 1的数组dp，其中dp[*i*]表示达到下标*i*的最小花费。由于可以选择下标0或1作为初始阶梯，因此有 dp[0] = dp[1] = 0。

当2 ≤ *i* ≤ *n* 时，可以从下标*i* – 1使用cost[*i* – 1]的花费达到下标*i*，或者从下标*i* – 2使用cost[*i* – 2]的花费达到下标*i*。为了使总花费最小， dp[*i*]应取上述两项的最小值，因此递归关系式如下：

dp[*i*] = min(dp[*i* – 1]+cost[*i* – 1], dp[*i* – 2]+cost[*i* – 2])

按照dp[1], dp[2], dp[3], …, dp[*n* – 1], dp[*n*]的顺序，自底向上（从小问题到大问题）依次计算数组dp中的每一项的值，最终得到的dp[n]即为达到楼层顶部的最小花费。

4. 工厂*s*生产的商品通过下图表示的运输网络运送到市场*t*，网络中边上的数值表示单位时间可以通过该边的商品数量的上限，求单位时间内从工厂*s*到市场*t*能运送的最大商品总量。要求给出求解过程（给出每条增广路径及其增加的流量值）。



（考试时可不画此图）

解：

增广路径#1：<*s*, *s*1, *v*1, *t*1, *t*>，流量+10；

增广路径#2：<*s*, *s*2, *v*5, *t*2, *t*>，流量+10；

增广路径#3：<*s*, *s*2, *v*2, *v*3, *v*4, *t*2, *t*>，流量+5；

增广路径#4：<*s*, *s*1, *v*2, *v*1, *t*1, *t*>，流量+4；

增广路径#5：<*s*, *s*2, *v*5, *v*3, *v*4, *t*2, *t*>，流量+3；

最大流大小为32，则单位时间内从工厂到市场所能运送的最大商品总量为32。