Homework 4

1. 假定你希望兑换外汇，你意识到与其直接兑换，不如进行多种外币的一系列兑换，最后兑换到你想要的那种外币，可能会获得更大收益。假定你可以交易 n 种不同的货币，编号为 $1,2,...,n$，兑换从 1 号货币开始，最终兑换为 n 号货币。对每两种货币 i 和 j，给定汇率 r_{ij}。意味着你如果有 d 个单位的货币 i，可以兑换 dr_{ij} 个单位的货币 j。进行一系列的交易需要支付一定的佣金，金额取决于交易的次数。令 c_k 表示 k 次交易需要支付的佣金。证明：如果对所有 $k=1,2,...,n$, $c_k=0$，那么寻找最优兑换序列的问题具有最优子结构。然后请证明：如果佣金 c_k 为任意值，那么问题不一定具有最优子结构。。

2. 设计一个高效的算法，对实数线上给定的一个点集 $\{x_1,x_2,...x_n\}$, 求一个单位长度闭区间的集合，包含所有给定的点，并要求此集合最小。证明你的算法是正确的。

3. 一位公司主席正在向 Stewart 教授咨询公司聚会方案。公司的内部结构关系是层次化的，即员工按主管-下属关系构成一棵树，根结点为公司主席。人事部按“宴会交际能力”为每个员工打分，分值为实数。为了使所有参加聚会的员工都感到愉快，主席不希望员工及其直接主管同时出席。

公司主席向 Stewart 教授提供公司结构树，采用左孩子右兄弟表示法（参见课本 10.4 节）描述。每个节点除了保存指针外，还保存员工的名字和宴会交际评分。设计算法，求宴会交际评分之和最大的宾客名单。分析算法复杂度。

4. 考虑用最少的硬币找 n 美分零钱的问题。假定每种硬币的面额都是整数。设计贪心算法求解找零问题，假定有 25 美分、10 美分、5 美分和 1 美分四种面额的硬币。证明你的算法能找到最优解。

1. 1) $C_k=0$ 时：

考虑前 n-1 种货币 假设知道了 1 兑 n-1 的方案

由 1 个单位 能兑 W_{1j} $(j=1,\cdots n-1)$ 的 j 货币.

则 兑 n 货币时 $W_{1n} = \max\limits_{1\le j\le n-1}\{W_{1j}r_{jn}, r_{1n}d\}$

故具有最优子结构

2) C_k 为任意值时

求解 n-1 兑换方案时会用到 k_{n-1}，而在求 n 兑换方案时会用到 k_n 依赖于 k_{n-1}，可能有一条不属于子问题的路径出现。

例如，真实最优解为 $1 \to y \to n$

而可能因为 c 很大导致搜索 w_{1y} 时

搜到的路径为 $1 \to x \to y$

$1 \to y$ 并不在子问题的解中.

2. 1) 取 $x_m^{(0)} = \min\{x_1, \cdots x_n\}$ 去掉所有小于 $x_m^{(0)}+1$ 的点

得到 $\{x_1^{(1)}, \cdots x_{n_1}^{(1)}\}$

2) 对得到的集合重复上述操作

直到得到 ϕ，解为 $\{[x_m^{(i)}, x_m^{(i)}+1] \mid i\}$

证明最优：

假设最优解为 S'

将 S' 中所有集合的下界移到区间包含的 x_i

的最小值，得到 S''，S'' 仍为最优解

1) 若 S'' 元素无交 则 $S = S''$，得证

2) 若 S'' 元素有交，则从下界最小的区间开始，

让有交的区间中下界较大的区间后移，保持

下界在 x_i 上，直到不相交，得到 S'''

则 $S''' = S$ 且 $|S'''| \leq |S''| = |S'|$ 故 S 为最优解

3. 对 $\forall$ 节点 p 为根构成的子树 设 $S_0[p]$ 为 p 不参加时最大评分和

$S_1[p]$ 为 p 参加时最大评分和

对于 p 的所有子节点 p_{ci}

$$\begin{cases} S_0[p] = \sum_i \max\{S_0[p_{ci}], S_1[p_{ci}]\} \\ S_1[p] = s[p] + \sum_i S_0[p_{ci}] \end{cases}$$

边界为所有的叶子节点 p_{leaf}

$S_0[p_{leaf}] = 0$

$S_1[p_{leaf}] = 1$

即可自顶向下递归，取 $\max\{S_0[p], S_1[p]\}$

当取 $S_1[p]$ 时记录 $attend[p]=1$ 帮助回溯

复杂度，访问 p 节点时，通过递归已知道所有的 $S_0[p_{ci}]$ $S_1[p_{ci}]$

故可在 $O(1)$ 得到 $s[p]$

故总复杂度为 $O(n)$

4. 每次取最大面额的硬币，当剩余的金额不足时取第二大面额，依此类推；由于总面额为整数且有1美分的硬币

显然能找到一个解，下面证明最优：

对 $\forall$ 面额 m 找零，若找零方案中25的数量 $x_1' < \lfloor \frac{m}{25} \rfloor = x_1$

则有 $(x_1 - x_1')\cdot 25 + m - x_1 \cdot 25$ 的硬币要用10, 5, 1美分找

币 $(x_1-x_1')25 + m - x_1\cdot 25$ 比 $m - x_1\cdot 25$ 多出了 $(x_1 - x_1')\cdot 25 \geq 25$

这些钱需要不少于(x_1-x_1')张10，5，1美分的钱找零
所以用x_1'张25美分找零比用x_1需要更多硬币
即取x_1最优。

同理，对剩下的钱也尽量多的找最大面额 此时最优
即为上述算法。