## 第一、二讲

计算下列算法复杂度

T(n)=3T(n/3)+nlgn

T(n)=3T(n/4)+nlgn

T(n)=9T(n/3)+n^1.8

T(n)=T(2n/3)+10000

T(n)=2T(sqrt(n))+lgn

## 第三讲 排序

* 给出随机快速排序的伪码
* 为什么快速排序比较快

### 第四、五讲 动态规划、贪心

* 矩阵连乘 a.描述最优子结构b.给出递规的定义c.给出算法伪码，自底向上贪心算法

### 第六讲 均摊分析

均摊分析：累计法，记账法，势能法。需要掌握

### 第七讲 堆

优先队列的复杂度：例如，如果用fibonacci heep，复杂度如何？不考如何实现

### 第八至十 图论

* 写BELLMAN-FORD的伪码，分析复杂度
* 给出拓扑排序算法和复杂度，有向无环图最短路径算法和复杂度。
* 给出了Prim算法。分析用普通数组，Binary Heap和Fibonacci Heap，对Prim算法的复杂度进行分析
* a.差分约束系统中AX<b A是n\*m b是m维向量，如何利用Bellman-Ford算法计算(描述) b.给出算法的改造，使得复杂度为O(mn) (描述)

### 第十一讲 字符串匹配

* 手工计算一个字符串的所有π值
* a.分析自动机的字符串匹配算法中，如果已知转移函数表，给出自动机的字符串匹配算法（伪码）,分析复杂度 b.分析自动机的字符串匹配算法中，转移函数的算法的复杂度。给出新的算法，复杂度是O(nX) X是输入字符集的大小（不需证明）

### 第十二十三 NP 近似算法

* 语言L属于{0,1}\*，说明L满足什么情况时为NPC
* 已知汉密尔顿问题为NPC，证明旅行商为NPC
* 旅行商问题的近似解法（似乎是）---应该是满足三角公式的吧？
* 怎样证明一个问题是NPC问题

### 其他

5，n个直径不同的螺丝，以及与这些螺丝相应的n个螺母，螺丝之间不能比直径大小，螺母之间也不能比较直径大小。描述把这n对螺丝螺母进行逐一匹配的算法并写出伪码，允许的复杂度为O(nlgn)

6，有m种不同面值的硬币，面额分别是d1,d2...dm

设计一个算法，使得总面额为n时，所用的硬币最少，并给出复杂度分析。

问什么时候该问题不可解（即无法取出硬币以凑成总面额为n）。

前面几题书上都有，简单说一下第五题和第六题的我自己的思路，不知道对不对仅供参考

第五题：存放螺丝的数组为p，存放螺母的数组为q

1，从p中的螺丝中随机取出一个与q中的所有螺母比较，把直径小于该螺丝的螺母放在数组q1中，把直径大于该螺丝的螺母放在数组q2中

2，把1中得到的直径与该螺丝相等的螺母取出来与所有的螺丝相比较，同样得到两个数组p1和p2。

3，得到一对正确的螺丝螺母匹配

4，对p1和q1递归调用本算法；对p2和q2递归调用本算法。

（终止条件为length[p]=length[q]=1，此时返回一对正确的匹配）

复杂度分析与快速排序有点像，excepted time为O(nlgn)

第六题：动态规划。贪心算法应该是不对的。

我用了一个数组p[i]来存放当面额为i时所需的最少硬币数目。

则p[0]=0，p[d[j]]=1（此时总面额正好等于某个硬币的面额）

p[i] = min(p[i-d[1]]+1，p[i-d[2]+1，...，p[i-d[m]+1]])

其他情况时（即不可解时）p[i]为无穷大

复杂度为O(nm)