习题3

1.填空

（1）利用分治递归计算Fibonacci数，时间复杂度非常高，主要原因是对函数重复调用，重复调用数量巨大，导致T（n）为指数阶。

（2）为了不重复计算子问题，主要的改进方法有两种，分别是备忘录方法和动态规划。

（3）矩阵连乘时，不同的乘法执行顺序，导致最终乘法次数差异很大。

（4）备忘录方法采用和分治递归相同的自顶向下的方式求解，而动态规划算法则是自底向上的方式求解。

（5）用备忘录方法求解LCS问题，和动态规划方法相比较，它的特点是算法控制结构不同和避免了无效计算。

（6）图像压缩的前提是图片中连续区域中像素点的灰度值时接近的，具体方法是分段存储。

（7）利用普通动态规划算法解0,1背包问题，主要的缺点是当背包容量很大时，计算时间较长和存在无效计算。

（8）最优二叉搜索树要解决的问题是找出一个二叉树搜索树，其平均搜索路长最小。

2.动态规划的基本步骤。

找出最优子结构，刻画其结构特征；

递归定义最优值（问题最优值和自问题最优值的关系）；

自底向上计算最优值（填表）；

根据计算最优值时得到的信息，构造最优解；

3. 对于计算A[i: j]的最优加括号计算次序，用一个最优值m[i: j]来表示该计算次序下的乘法次数， Ai的维数为pi−1×pi ， A1A2…An的维数表示为p0p1…pn。给出m[i: j]和子问题的递推关系式，并解释式中各项的含义。

m[i,k]和m[k+1,j]分别是已求得的第i到第k项和第k+1到第j项最优计算次数，为i到k项相乘的结果与k+1到j项相乘的结果相乘需要的乘法次数。

4.动态规划的适用条件。

用动态规划解决的问题必须有一下两个性质：

最优子结构和自问题重叠。

5. 最长单调递增子序列

设计一个时间的算法，找出由n个数组成的序列的最长单调递增子序列。

方法一：设l[i]表示以第i个元素结尾的最长递增子序列的长度，则

l[i]=max{l[j] | j<i,a[j]<a[i]}+1；

方法二：设原序列为A将其按升序重排为B，类似LCS的方法所求即为所得。

6.长江游乐俱乐部在长江上设置了n个游艇出租站，游客可以在这些游艇出租站用游艇，并在下游任何一个游艇出租站归还游艇，游艇出租站i到j之间的租金是r (i,j)，其中1<=i<j<=n。试设计一个[算法](http://lib.csdn.net/base/datastructure)，计算从游艇出租站1到出租站n所需的最少租金。

略，类似矩阵连乘问题。