HW7

王嵘晟　PB1711614

1.

最长简单路径从点 s 开始，一定经过 s 的某条边，且在到达 t 前不会再次路过已经经过的点。所以状态转移方程为：

$$LogestPath(G,s,t) = MAX(LogestPath(G',s',t)) + 1, G' = G - \{s\}, ss' \in E$$

当 s=t 时，最长简单路径长度为 0，当 S≠t 时，根据状态转移方程由广度优先搜索可以寻找到最长的简单路径。由于这样需要遍历整张图才能找到最长简单路径，所以时间复杂度为 $O(|V|^2 2^{|V|})$。

3.

由于员工及其直接主管不同时出席，所以如果树根 r，即主席包含在一个最优解中，则其子辈不在最优解中，孙辈一定在最优解中。只需求解根在 r 孙辈的最优子问题。同样如果树根不在最优解中，则其子辈一定在最优解中。只需求解根在 r 子辈的最优子问题。所以首先建立由顶点索引的表 A，将根在该结点上的所有子树中员工的“宴会交际能力”由大到小排序。然后建立另一个表 B，B[n] 表示顶点 n 上的子树的宴会交集评分之和最大的宾客名单。左孩子右兄弟树为 T。对于每个叶子结点 L，如果 L 的交际能力 $>$0，则 B[L]=L,A[L]=L.ability。否则 B[L]=$\emptyset$，A[L]=0。通过迭代可以求解子问题所在结点的父结点的子问题，对结点 x，有转移方程：

$$A[x] = MIN(\Sigma_{y\ is\ x's\ child} A[y], \Sigma_{y\ is\ x's\ grandchild} A[y])$$

当 n 为员工数量时，时间复杂度 $O(n^2)$。

Algorithm 1 *

2. 动态规划求解 0-1 背包

function Dynamic-0-1-Pack(v, w, n, W)
    for $w \quad from \quad 0 \quad to \quad W$ **do**
        $c[0, w] = 0$
    end for
    for $i \quad from \quad 1 \quad to \quad n$ **do**
        $c[i, 0] = 0$
        for $w \quad from \quad 1 \quad to \quad W$ **do**
            if $w_i \leq W$ **then**
                if $v_i + c[i-1, w-w_i] \leq c[i-1, w]$ **then**
                    $c[i, w] = v_i + c[i-1, w-w_i]$
                else
                    $c[i, w] = c[i-1, w]$
                end if
            else
                $c[i, w] = c[i-1, w]$
            end if
        end for
    end for
    return $c[n, w]$
end function

4.

首先对这个点集使用归并排序有小到大排序，设排序后的点集为 $y_1, y_2, ...y_n$，第一个区间为 $[y_1, y_1 + 1]$, 如果 y_i 是不包含任何现有的区间内点的最左端点，则下一个区间为 $[y_i, y_i + 1]$。使用贪心算法，可以保证集合中每个元素必然被包含在一个区间内，所以这样处理一定可以得到最优解。这种算法的时间复杂度为 $O(nlgn)$

5.

如果要找零的 $n \geq 25$，则找零 25 美分，n=n-25，直到 n 比 25 要小，若此时 $n \geq 10$，则继续找零 10 美分，以此类推可以保证找零用硬币最少。不采用这种方法找零，找零的硬币数一定大于等于这种方法，所以这种方法为最优的。