**Monday Homework (due 9am on Wednesday)**

**Data Structures (Robert Tarjan)**

**Problem 1 (5 points)**

(a) Give the AVL tree resulting from the insertion of a, b, c, d, e, f, g, h, i, j (in that order) into an initially empty tree.

(b) Give the red-black tree resulting from the sequence of insertions in (a) into an empty tree.

(c) Give the splay tree resulting from the sequence of insertions in (a) into an empty tree.

**CORRECTED: Problem 2 (15 points)**

(a) Show that any rank-balanced tree can be converted into a red-black tree by giving each of its nodes a suitable color.

(b) Give a red-black tree that CANNOT be converted into a rank-balanced tree via any rank assignment to its nodes.

**Large Data Sets (Giuseppe Italiano)**

**Problem 1 (10 points)**  
  
If a tree *T* is represented as an unordered collection of edges (i.e., the adjacency list representation of *T* is not available), what’s the I/O complexity of computing an Euler Tour of *T*?

**Problem 2 (10 points)**  
  
Given a tree *T*, rooted at a node *r*, the depth of a node *v* is defined as the number of edges on the path from *r* to *v* in *T*. Design an I/O-efficient algorithm to compute the depth of each node in *T*.