1. (a) Give an AVL tree of height 4 with the least possible number of nodes.
(b) Suppose we allocate 8 bits for the height counters in each node of an AVL tree. What is the smallest number of nodes in the tree that might cause the height of the root to overflow? (It is sufficient to give an estimate for the order of magnitude.)

2. Show that any binary search tree can be transformed into any other binary search tree (for the same keys) just by performing rotations. Estimate the number of rotation needed in the worst case. (Here the trees are of course not assumed to be balanced.)

   Hint: first show that any tree can be transformed into a right-going chain, i.e. a tree where no node has a left child.

3. Assume a binary tree has $m$ leaves $l_1, \ldots, l_m$, and the depths of the leaves are $d_1, \ldots, d_m$. Show that

   $$\sum_{i=1}^{m} 2^{-d_i} \leq 1.$$

   When does this hold as equality? (This result is a special case of Kraft’s inequality, which is central in coding theory.)