Algorithms — Homework 6
Balanced Trees

Due: October 20.

1) Argue that since sorting $n$ elements takes $\Omega(n \log n)$ time in the worst case in the comparison model, any comparison-based algorithm for constructing a binary search tree from an arbitrary list of $n$ elements takes $\Omega(n \log n)$ time in the worst case.

2) You have given a sorted array $A$ with $n$ numbers. Describe a linear time algorithm that builds a balanced BST storing all elements in $A$.

3) What is the largest possible number of internal nodes in a red-black tree with black-height $k$? What is the smallest possible number?