January 2010
Preliminary Exams

Design & Analysis of Algorithms (Questions 5 – 8)

Problem 5

Design an $O(\log n)$ algorithm that determines whether a red-black tree with n keys stores any keys within a certain (closed) interval. That is, the input to the algorithm is a red-black tree T and two keys, l and r ($l \leq r$). If T has at least one key k such that $l \leq k \leq r$, then the algorithm returns *true*, otherwise it returns *false*.

Problem 6

Suppose we are given a sequence S of n elements, each of which is an integer in the range $[0, n^2 - 1]$. Describe a simple method for sorting S in $O(n)$ time.

Problem 7

Suppose you are given a diagram of a telephone network, which is a graph G whose vertices represent switching centers, and whose edges represent communications lines between two centers. The edges are marked by their bandwidth. The bandwidth of a path is the bandwidth of its lowest bandwidth edge. Describe an algorithm that, given a diagram and two switching centers a and b, will output the maximum bandwidth of a path between a and b. (Just report the maximum bandwidth; you do not have to give the actual path). Analyze the running time of your algorithm.

Problem 8

Suppose an oracle has given you a magic computer, C, that when given any undirected graph G will tell you in one step if G has a Hamiltonian cycle or not. Show how to use C to construct an actual Hamiltonian cycle in any undirected graph G. How many calls do you need to make to C in the worst case in order to do this? Recall that a cycle in G is *Hamiltonian* if it visits all vertices of G and each vertex exactly once.