CS 473: Algorithms, Fall 2010
HW 7 (due Tuesday, October 26)

This homework contains four problems. **Read the instructions for submitting homework on the course webpage**. In particular, *make sure* that you write the solutions for the problems on separate sheets of paper; the sheets for each problem should be stapled together. Write your name and netid on each sheet.

Collaboration Policy: For this home work, Problems 1-3 can be worked in groups of up to 3 students each.

Problem 0 should be answered in Compass as part of the assessment HW6-Online and should be done individually.

0. (10 pts) HW7-Online on Compass.

1. (25 pts) Let $G = (V, E)$ be an undirected graph with edge costs c_e. Given an integer k where $1 \leq k \leq n-1$, describe an algorithm that finds the minimum-cost forest in G that contains k edges.

2. (25 pts) Recall from class the idea of Path-Compression in the Union-Find data structure for maintaining disjoint sets. Prove that if Path-Compression is used then the total time for k operations in which all the `union` operations precede all the `find` operations, is $O(k)$. (The cost of initializing the data structure via `makeUnionFind` is not part of the cost of the k operations). Note that the running time is independent of the number of elements n.

3. Consider the randomized selection algorithm discussed in class.

   - (15 pts) We saw the the expected running time of the algorithm is $O(n)$ on an array of size n. Intuitively this is because the size of the array in each recursive call goes down by a constant factor since a random pivot is likely to divide the array into roughly equal sized arrays. Formalize this intuition and show that the expected number of recursive calls of randomized selection on an array of size n is $O(\log n)$. It is sufficient to prove this for selecting the median. *Hint:* Modify the recurrence for the running time to count the number of recursive calls.
   - (25 pts) Let $A_1, A_2, \ldots, A_k$ be k *sorted* arrays where the size of A_i is n_i. Let $n = \sum_i n_i$ be the total number of elements in all the arrays (assume they are distinct numbers). Describe a randomized algorithm to find the median of the numbers in the combined set of arrays in $O(k \log^2 n)$ expected time. *Hint:* Adapt the randomized selection algorithm and make use of the fact that the arrays are sorted.

4. Extra credit (15 pts): Prove that the recurrence for randomized Quick Sort is bounded by $O(n \log n)$.