CS 473: Algorithms, Fall 2010
HW 4 (due Tuesday, September 28th)

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 HW2-Online and should be done individually.

1. (30 pts) You are given an array A of n distinct integers, and an integer k such that $1 \leq k \leq n$. The *square distance* between a pair of integers x, y is defined as the quantity $(x-y)^2$. Your goal is to design an $O(n)$ time algorithm to find k elements in A with the smallest square distance to the median (i.e. the element of rank $\lfloor n/2 \rfloor$ in A). For instance, if $A = [9, 5, -3, 1, -2]$ and $k = 2$, then the median element is 1, and the 2 elements in A with the smallest square distance to the median are $\{1, -2\}$. If $k = 3$, then you can output either $\{1, -2, -3\}$ or $\{1, -2, 5\}$. *Hint:* Use the linear time Selection algorithm.

2. (30 pts) Describe an $O(n \log n)$ time algorithm to compute the longest decreasing subsequence of a given sequence of n numbers. (*Hint*: use an appropriate data structure to speed up the dynamic programming algorithm.)

3. (30 pts) Let T be rooted tree with integer lengths on its edges (they can be positive, negative or zero). Describe an algorithm that finds among all paths from a node in T to one of its descendants the one with the smallest length. The instance of the problem given below has the smallest path $E - F - H$ of length -9.