CSE 326b DATA STRUCTURES
HOMEWORK 3

Due: **Friday, Feb. 1, 2008** at the beginning of class. As usual, please put your quiz section (BA, BB) in addition to your name at the top of your homework, and nicely typeset solutions (LaTeX, Word, etc.) are encouraged.

Problem 1. Leftist Heaps

Leftist heaps support fast merges. Here is a chance to discover whether this is true or not....

(a) Show the result of inserting keys 1 to 15 in order (i.e. 1 first, then 2 second, then 3 third, etc.) into an initially empty leftist heap. Use the leftist heap insert (i.e. merge) algorithm at each step. You don't need to show each step for this process, but be warned that if all you write down is the final answer and you get it wrong, it will be hard to award any partial credit. (Yes, this is a bit of busy work, but it should help get you started with part (b) of the problem.)

(b) Prove or disprove: A perfectly balanced tree forms if keys 1 to $2^k - 1$ are inserted in order (again this means 1 first, then 2 etc) into an initially empty leftist heap. k is a positive integer. (Hints: induction; you have already worked through the base case above.)

Problem 2. Skew Heaps

(a) Weiss 6.26. You only need to show the final result, but note that if you do this it will be hard to award partial credit if the final result has problems.

Problem 3. Binomial Trees

A binomial tree of height 0, B_0, is a one-node tree. A binomial tree of height k, B_k is formed by attaching a binomial tree, B_{k-1} to the root of another binomial tree another binomial tree B_{k-1}. (These are the same definitions as in Weiss.)

(a) Weiss 6.32.

(b) Prove that a binomial tree B_k has 2^k nodes.