CMPS 101
Final Review Problems

1. Let T be a binary tree, and let $n(T)$ and $h(T)$ denote its number of nodes and height, respectively. Show that $h(T) \geq \lfloor \lg(n(T)) \rfloor$. (Hint: this was proved in the solutions to hw6.)

2. Trace HeapSort on the following arrays
   a. (9, 3, 5, 4, 8, 2, 5, 10, 12, 2, 7, 4)
   b. (5, 3, 7, 1, 10, 12, 19, 24, 5, 7, 2, 6)
   c. (9, 8, 7, 6, 5, 4, 3, 2, 1)

3. Draw the Binary Search Tree resulting from inserting the keys: 5 8 3 4 6 1 9 2 7 (in that order) into an initially empty tree. Write pseudo-code for the following recursive algorithms, and write their output when run on this tree.
   a. InOrderTreeWalk()
   b. PreOrderTreeWalk()
   c. PostOrderTreeWalk()

Note: Some of the topics represented by the following problems my not be covered by end of business Tuesday 8/11/09. If that is the case, those topics will not appear on the final exam.

4. The predecessor of a node *x* in a Binary Search Tree is defined to be the node which is printed immediately before *x* in an InOrderTreeWalk(). Let *T* be a Binary Search Tree and let *x* be a node in *T*. Suppose *x* has no left child, and *x* has a predecessor *y*. State a characterization of the predecessor *y* similar to the characterization of successor in problem 2 of hw8. Write an algorithm called TreePredecessor() that returns the predecessor of a non-nil node *x*, if it exists, and returns nil otherwise.

5. State the following properties:
   a. The Binary Search Tree Properties:
   b. The Red-Black Tree Properties:

6. Let *x* be a node in a Red-Black tree, let $\mathrm{bh}(x)$ denote the black-height of *x*, and let $N(x)$ denote the number of internal (i.e. non nil) nodes in the subtree rooted at *x*. Show that $N(x) \geq 2^{\mathrm{bh}(x)} - 1$

7. Prove that any Red-Black tree on *n* nodes and with height *h* satisfies $h \leq 2\lg(n+1)$. (Hint: use the result of the preceding problem.)

8. Let *x* be a node in a red-black tree. Show that the longest path from *x* to a descendent leaf has length at most twice that of a shortest such path.

9. Draw the Red-Black tree which results from inserting the keys 5, 4, 1, 3, 2 (in order) into an initially empty tree. Draw all intermediate trees in this process.