CS W3137 Hmwk 3. DUE: Tues., March 11 in class

Non-Programming Problems, 3 points each. Answers to these questions will be submitted in class. Programming problems will be submitted electronically. See instructions on class home page. NOTE: these exercises are from the WEISS textbook.

1. Give a formal proof using induction that the number of leaf nodes in a perfect binary tree of height K is 2^K.

2. Give a formal proof using induction that the number of interior nodes in a perfect binary tree of height K is $2^K - 1$.

3. Give a formal proof using induction that the number of nodes in a perfect *trinary* tree of height K is $\frac{3^{K+1}-1}{2}$.

4. Exercise 4.4

5. Exercise 4.6

6. Exercise 4.8

7. Exercise 4.31b

8. Exercise 4.31c

9. Exercise 4.32

10. Exercise 4.33

11. Exercise 4.37

12. Exercise 4.46

13. Given a binary tree T, devise a method to output the tree to a file so it can be read in again by another program. Outline briefly how the file may be read into the correct tree structure. Note: You MAY NOT use the Java serializable class to do this.

14. Prove that given a preorder and a postorder traversal of a binary tree, the tree cannot be uniquely reconstructed.

15. The preorder traversal of a binary tree is "A B C D E F G H I" and inorder traversal is "C D E B F A I H G". Draw the tree.

Programming Problem (55 points)

1. You will create a program that will read in a legal infix arithmetic expression and output an expression tree for the infix expression. The infix input string can include integers, the operator set of +,-,*,/,^ and parentheses.

   What you need to do:

   (a) (5 pts) Create an input text area in a GUI to allow the user to type in the infix expression.

   (b) (20 pts) Create a button to convert the infix to postfix format, and print the postfix out in another text area of the GUI.

   (c) (20 pts) Create a button to convert the postfix expression into an expression tree, and display the expression tree in the GUI.

   (d) (5 pts) Create a button to traverse the expression tree to evaluate the expression's value and output the value in the GUI.

   (e) (5 pts) Create a button to traverse the expression tree and output in another text area of the GUI a fully parenthesized version of the expression in infix form.

   Extra credit (4 pts): Have your program accept legal infix expressions with a unitary minus operator such as -3*(5-3)+(-5). You can see what is legal by running the Unix calculator program *bc* and seeing if it accepts your input.