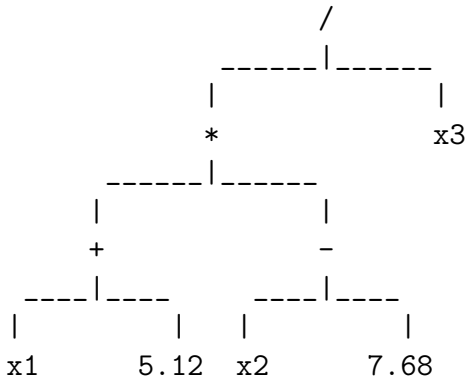
# CIS*2520 Data Structures

Fall 2014

**Assignment 3**

(1) (50%) Write a C program that takes as input a fully parenthesized, arithmetic expression of binary operators $+, -, *, /$, and converts the expression into a binary expression tree. Your program shall allow for the leaves not only to store floating values but also to store variables of the form x1, x2, x3, and so on, which are initially 0.0 and can be updated interactively by the user. For example, expression $(((x1 + 5.12) * (x2 - 7.68))/x3)$ will be converted into a binary expression tree like:




Your program should then show a menu with the following options:

1. Display
2. Preorder
3. Inorder
4. Postorder
5. Update
6. Calculate
7. Exit

Description:

- When option 1 is selected, your program should display the tree in some way so that the tree can be visualized, and also print the name/value of variables (if any).
- If an option of 2, 3 or 4 is selected, your program should print the expression by the corresponding traversal order (Note: no parentheses for preorder and postorder traversal but fully parenthesized for inorder traversal).
- Option 5 requires further input from user. A pair of input, namely,
  *variable_name, new_value*
  will be provided interactively and your program should search for the *variable_name* and replace its value by *new_value*.

- Arithmetic calculation is invoked by option 6 which shall display the calculation result.
- Option 7 terminates your program.

(2) (50%) Write a C program. It reads 200 2-digit integers from text file "f.dat" and stores the integers in an array of 20 rows and 10 columns. The program treats each row of the array as an object, with the sum of the first three integers being the key, and the other seven integers being the information content. The program then creates a heap with a node containing an object. You are required to use an array representation of heap, and apply the parental node downheap algorithm. The program finally displays the heap as a $20 \times 10$ array, a row for an object.

**Due time: 06:00, Monday, Nov 3, 2014**.