SUBMISSION AND GUIDELINES ARE SAME AS ASSIGNMENT 2

Additional Requirements are:

- Your program should accept a fully parenthesized, arithmetic expression as command line argument
 Example: treecalc "((x1 + 5.15) * (x2 7.68))"
 Note: you have to use a pair of single/double quotation marks to enclose the expression The input expression may involve blank characters, for example: ((x1 + 5.15) * (x2 7.68)) is equivalent to ((x1+5.15)*(x2-7.68))
- (2) You are NOT expected to handle unary operators such as negation. You only need to handle four operations: + * /.
- (3) Make your program display a welcome banner when started. This welcome banner should contain your full name and student ID. After that, your program MUST display a prompt when it is ready for new input. The prompt must the following string: "calc/> " Make sure to include a blank space right after the prompt.
- (4) Option 1: Display your tree in any form you please. Make sure that the tree structure is clearly visible. Be creative!
- (5) Option 2: Print your tree in preorder traversal on the next line. Use one and only one blank space between each element of your tree.
- (6) Option 3: Print your tree in inorder traversal on the next line. Use one and only one blank space between each element of your tree, and between parenthesizes.
- (7) Option 4: Print your tree in postorder traversal on the next line. Use one and only one blank space between each element of your tree.
- (8) Option 5: The syntax you MUST use for this option is: 'calc/> 5 x1, 20.5' Note the location of the space between the option number and the option arguments. Note also that there is no space between the variable name, the comma and the variable new value. You are not required to support variable name change.
- (9) Option 6: Compute the result of the expression down to 4 digits of precision (aka: 4 digits after the dot). Example: 3.1415
- (10) Option 7: Terminate your program.