Expression Tree

Evaluate an expression represented by a String.

Expression can contain parentheses, you can assume parentheses are well-matched.

For simplicity, you can assume only binary operations allowed are +, -, *, and /.

Arithmetic Expressions can be written in one of three forms:

Infix Notation: Operators are written between the operands they operate on,

e.g.   5*(3+4) - 6

Your program should read input strings from file.

input file format:

e.g.

5*(3+4) - 6
10 - 18/-2
11/ 5 * (3+3)
1/3 * 5

You will have 2 output files:

1. Write all final output result to file "project1_output.txt":
    output file format:

    e.g.

    5 * (3 + 4) – 6 = 29
    10 - 18 / -2 = 19
    11 / 5 * (3 + 3) = 13.2
    1 / 3 * 5 = 1.67

2. For each expression, after you construct the expression tree, print the preorder of that tree to file "project1_debug.txt"

    e.g.

    - * 5 + 3 4 6
    - 10 / 18 -2
    * / 11 5 + 3 3
    * / 1 3 5

Note:

1. Your answer should round to 2 decimal places.
2. You should solve this problem using binary tree as we described in class.
3. Put all your java classes in 1 file:
    Filename format:
            Firstname_Lastname_CUNYID#_Project2.java
4. Your source file should be uploaded to blackboard.
5. Make sure your java code can be tested using Java 8.