605.202.31

Data Structures

Lab1 – Stack

Choice 2

Problem

Write a Java Program to convert
   1. A Prefix string to Postfix
   2. A Postfix string to Prefix
   3. A infix string to Prefix
   4. A Prefix string to infix.
   5. A Postfix string to infix.

Test Strings:
Postfix to Infix & Prefix
   a. AB + C –
   b. ABC +-
   c. AB -C + DEF -+^
   d. ABCDE -+ ^*EF*-

Prefix String to Infix & Postfix
   a. ++A - * ^BCD/ + EF * GHI
   b. +-^ABC * D ** EFG

Infix to Prefix
   a. (A + B) * (C ^(D-E) + F) – G
   b. A + (((B-C) * (D-E) + F)/G) ^ (H-J)

In addition to the above test expression, you may need to include few more expressions of your own and show converted results.

In your analysis, be sure to discuss the implementation you choose and why, why a stack makes sense. Consider a recursive solution (you do not need to implement recursion) and compare it to your iterative solution. Is one better than the other? Why? Tell us what you learned and what you would do differently.

Be sure to review the programming assignment guidelines, including the formatting requirements for the analysis. You may not use library functions. You must write your own code, in particular you must write the stack code. Be sure to include the stack source code in your submission. You must read and write from named files.