ICSI213 Data Structures

Notice that students are expected to start the lab as soon as the description is available and seek feedback during the lab. Labs are contiguous study of the lecture or used as stepping-stones for the projects. Skipping lab activities would impact the learning significantly.

Lab 02 Three languages for algebraic expressions

For this lab, you will study and work on the problems included on the next page.

There are three languages for algebraic expressions:

- **Infix**: An operator appears between its operands.
- **Prefix**: An operator appears before its operands.
- **Postfix**: An operator appears after its operands.

Infix	Prefix	Postfix
a + b	+ a b	a b +
a + b * c	+ a * b c	a b c * +

In all three versions, the operands occur in the same order, and just the operators must be moved to keep the meaning correct. A fully parenthesized infix algebraic expression is an infix algebraic expression where every operator and its operands are contained in parentheses as seen in the following examples:

Not Fully Parenthesized	Fully Parenthesized
a	(a)
a + b	(a + b)
a + b * c	(a + (b * c))
(a + b) * c	((a + b) * c)
a + (b * c) - d	((a + (b * c)) – d)

To convert **a fully parenthesized infix** expression to a prefix,

- Move each operator to the position marked by its corresponding open parenthesis
- Remove the parentheses
- Example:
  - Infix form: ((a + b) * c)
  - Prefix form: * + a b c

To convert **a fully parenthesized infix** expression to a postfix,

- Move each operator to the position marked by its corresponding closing parenthesis
- Remove the parentheses
- Example:
  - Infix form: ((a + b) * c)
  - Postfix form: a b + c *

For this lab, you will work with a partner to solve the following problems. Submit one PDF file with all solutions included.

Question 1	10 points
Question 2	10 points
Question 3	10 points
Question 4	10 points
Total/ 40	40 points
Total/ 5	5 points

Problems:

Convert the following infix expressions into fully parenthesized infix expressions. And then,

A + B * C + D

(A + B) * (C + D)

A * B + C * D

A + B + C + D

convert each fully parenthesized infix expression into to prefix and postfix. Compare your results with others.