Name/PID: _____

COMP520: Written Assignment 2

Q1. **Stratified Grammars**. Build an LL(1) Stratified Grammar with the following specification: Four arithmetic terminals exist (num, +, -, x). Multiply (denoted by an x) is a higher precedence than addition and subtraction. The - symbol can represent both a unary negation and subtraction. Unary negation is right associative and has a higher precedence than multiplication.
Ensure expressions will be evaluated with proper precedence and no left recursion.

Q2. **Precedence Levels in Stratified Grammars**. Assume two (or more) operations happen at the same precedence level in a stratified grammar. When building the ASTs as described in lecture, will the same-precedence operations happen in left-to-right order, or right-to-left order? Briefly explain in max two sentences.

Q3. **ASTs.** Show the AST for 2 - 3 + -4 - - -5. Use BinExpr, UnaryExpr, and NumExpr nodes. At the leaf nodes, show the underlying text for that node. You can use PA2 if you want to check your work.

Q4. **PA2 Concepts**. Describe how you would modify the simpleAST example on the class website to build Expr ASTs using BinExpr, UnaryExpr and NumExpr nodes? Describe your extensions to the scanner, parser, AbstractSyntaxTrees, and visitors. It's not necessary to write out complete code, but if you are interested you can extend the simpleAST example to try it out.

a. Updates to Scanner:
Hint: These are changes needed to scan any new tokens / account for new TokenTypes.

b. Updates to Parser:
Hint: These are changes needed to recognize new syntax.

c. Updates to AbstractSyntaxTrees Package:
Hint: These are changes needed to construct the new AST nodes.

d. Updates to AbstractSyntaxTrees Visitors:
Hint: These are changes needed to traverse the new AST nodes.