Assignments are due on the due date before class and have to include this cover page. Plagiarism in assignment answers will not be tolerated. By submitting their answers to this assignment, the authors named above declare that its content is their original work and that they did not use any sources for its preparation other than the class notes, the textbook, and ones explicitly acknowledged in the answers. Any suspected act of plagiarism will be reported to the Faculty's Academic Integrity Officer and possibly to the Senate Discipline Committee. The penalty for academic dishonesty may range from failing the course to expulsion from the university, in accordance with Dalhousie University's regulations regarding academic integrity.
Question 1 (15 marks) Consider the following grammar. (As usual, capital letters are non-terminals, lowercase letters are terminals.)

\[
\begin{align*}
\text{Statement} & \rightarrow \text{Assignment} \\
\text{Statement} & \rightarrow \text{SubroutineCall} \\
\text{Assignment} & \rightarrow id \ := \ Expression \\
\text{SubroutineCall} & \rightarrow id ( \text{ArgumentList} ) \\
\text{Expression} & \rightarrow \text{Primary \ ExpressionTail} \\
\text{ExpressionTail} & \rightarrow \text{Operation \ Expression} \\
\text{ExpressionTail} & \rightarrow \epsilon \\
\text{Primary} & \rightarrow id \\
\text{Primary} & \rightarrow \text{SubroutineCall} \\
\text{Primary} & \rightarrow ( \text{Expression} ) \\
\text{Operation} & \rightarrow + \\
\text{Operation} & \rightarrow - \\
\text{Operation} & \rightarrow * \\
\text{Operation} & \rightarrow / \\
\text{ArgumentList} & \rightarrow \text{Expression \ ArgumentListTail} \\
\text{ArgumentListTail} & \rightarrow , \text{ArgumentList} \\
\text{ArgumentListTail} & \rightarrow \epsilon
\end{align*}
\]

(a) Prove that the grammar is not LL(1).
(b) Modify the grammar so that it is LL(1).
(c) Provide a recursive descent parser for the language produced by the grammar in (b). Present this parser in pseudo code, C, C++, Java, Ruby, Python or Scheme.

Question 2 (15 marks) For each of the following languages, provide a grammar that generates it, an intuitive explanation why this grammar generates this language, and a graphical representation of a push-down automaton that recognizes this language.

(a) The language of properly nested sets of parentheses over the alphabet \{ (, ) \}. Note that the string \((()())()\) belongs to this language, while the string \((()())()\) does not because the third closing parenthesis does not have a matching opening parenthesis.

(b) The language of all binary strings with the same number of 0s and 1s.

(c) The language of all strings over the alphabet \{a, b, c\} that are not palindromes, that is, are not of the form \(\sigma \alpha \sigma^R\), where \(\sigma^R\) is the reversal of string \(\sigma\) and \(\alpha \in \{a, b, c, \epsilon\}\). Thus, the string \(abb\) belongs to this language, while the string \(abcba\) does not.