Assignment 4
COMP 330 Winter 2021 McGill University

Due Date: 11$^{\text{th}}$ March 2021

18th February 2021

There are **5** questions for credit, one extra question to sharpen your udnerstanding and one for your spiritual growth. The homework should be submitted through myCourses as a pdf in a single file.

Question 1[25 points] A sequence of parentheses is a sequence of `(` and `)` symbols or the empty sequence. Such a sequence is said to be *balanced* if it has an equal number of `(` and `)` symbols **and** in *every* prefix of the sequence the number of left parentheses is greater than or equal to the number of right parentheses. Thus `((())())` is balanced but, for example, `())(` is not balanced even though there are an equal number of left and right parentheses. The empty sequence is considered to be balanced.

Consider the grammar

$$S \to (S)|SS|\varepsilon.$$

This grammar is *claimed to* generate balanced parentheses.

1. Prove that any string generated by this grammar is a properly balanced sequence of parentheses. [10]
2. Prove that every sequence of balanced parentheses can be generated by this grammar. [15]

Question 2[15 points] Consider the following context-free grammar

$$S \longrightarrow aS \mid aSbS \mid \varepsilon$$

This grammar generates all strings where *in every prefix* the number of a's is greater than or equal to the number of b's. Show that this grammar is ambiguous by giving an example of a string and showing that it has two different derivations.

Question 3[15 points] We define the language $PAREN_2$ inductively as follows:

1. $\varepsilon \in PAREN_2$,
2. if $x \in PAREN_2$ then so are (x) and $[x]$,
3. if x and y are both in $PAREN_2$ then so is xy.

Describe a PDA for this language which accepts by empty stack. Give all the transitions.

Question 4[20 points] Consider the language $\{a^n b^m c^p | n \leq p \textbf{ or } m \leq p\}$. Show that this is context free by giving a grammar. You need not give a *formal* proof that your grammar is correct but you must explain, at least briefly, why it works.

Question 5[25 points] A *linear* grammar is one in which every rule has exactly one terminal and one non-terminal on the right hand side or just a single terminal on the right hand side or the empty string. Here is an example

$$S \to aS|a|bB; \quad B \to bB|b.$$

1. Prove that any regular language can be generated by a linear grammar. I will be happy if you show me how to construct a grammar from a DFA, if your construction is clear enough you can skip the straightforward proof that the language generated by the grammar and the language recognized by the DFA are the same.

2. Is every language generated by a linear grammar regular? If your answer is "yes" you must give a proof. If your answer is "no" you must give an example.

Extra Question[0 points] Prove that the grammar in Question 2 generates only strings with the stated property and *all* strings with the stated property.

Spiritual growth[0 points] Show that the language $L = \{a^n b^n | n \geq 0\} \cup \{a^n b^{2n} | n \geq 0\}$ is context free but is not accepted by any DPDA.