Theory of Computation '24
Problem Set 3

Notations. Let $\Sigma = \{a, b\}$. For $w \in \Sigma^\star$, let $|w|$ denote the length of w. Let $\#_a(w)$ denote the number of as in w and let $\#_b(w)$ denote the number of bs in w.

1. Show that if L is a context-free language, then:
   (a) $L^{\mathcal{R}}$, that is the language containing the reverse of the strings in L, is also context-free
   (b) L^*, Kleene Star of L, is also context-free
   (c) $f(L)$ is also context-free, where f is a homomorphism from $\Sigma \to \Gamma^*$ mapping each character of a string $w \in L$ to a string in some other alphabet Γ

2. **Definition 1** *A grammar (V,Σ,R,S) is said to be right linear if* $\forall\, H \in R$, H *is of the form:*
   - $Y \to \alpha X$, *where* $Y, X \in V$ *and* $\alpha \in \Sigma^*$
   - $Y \to \alpha$, *where* $Y \in V$ *and* $\alpha \in \Sigma^*$

   Definition 2 *A grammar (V,Σ,R,S) is said to be left linear if* $\forall\, H \in R$, H *is of the form:*
   - $Y \to X\alpha$, *where* $Y, X \in V$ *and* $\alpha \in \Sigma^*$
   - $Y \to \alpha$, *where* $Y \in V$ *and* $\alpha \in \Sigma^*$

   Based on these definitions, prove the following statements:
   (a) Given a regular language L, show that there exists both a right linear grammar T and a left linear grammar U such that $L(T) = L(U) = L$
   (b) Given a right linear grammar T, show that $L(T)$ is regular
   (c) Given a left linear grammar U, show that $L(U)$ is regular

3. **Balanced Parentheses.** A string defined on the alphabet $\Sigma = \{(,)\}$ is called a *balanced parenthesization* if the number of (is equal to the number of) and for every prefix, the number of) is not greater than the number of (. Give CGFs for balanced parenthesizations which satisfy the following two additional properties.
   (a) contain an even number of opening parenthesis

(b) do not contain $(())$ as a subword

(c) Consider the following CFG for the language of all balanced parenthesizations

$$X \leftarrow X(X)|\varepsilon$$

Prove rigorously that the above CFG is unambiguous

4. Consider the following algorithm which takes a grammar $G = (V, \Sigma, P, S)$ as input and returns `True` or `False`. As usual, V is the set of non-terminals, Σ is the alphabet, P is the set of production rules, and $S \in V$ is the initial non-terminal.

Algorithm 1 Alg(V, Σ, P, S)

$T \leftarrow \emptyset$.
while P contains a production rule of the form $A \rightarrow \alpha$, where $A \notin T$ and $\alpha \in T^*$ **do**
  $T \leftarrow T \cup \{A\}$.
end while
If $S \in T$ then return `True`, else return `False`.

What interesting property of the grammar G is the above algorithm testing ? Prove your answer rigorously. This means, you have to prove the following - the algorithm returns True if and only if the said property is true for the given grammar.

5. Prove that the following are not context-free:

(a) $\{ww \mid w \in \{a, b\}^*\}$

(b) $\{0^i 1^{i^2} \mid i \geq 0\}$

(c) $\{x \in \{0, 1\}^\star || x| = n^2$ for some $n \in \mathbb{N}\}$ [**Hint:** Use closure properties]