COMP 330 Fall 2023
Assignment 4
Due date: 3^{rd} Nov 2023

Posted on October 20^{th} 2023

This is the fourth of 6 assignments. There are **5** questions for credit. The homework is due on `CrowdMark` **at 11:59pm**.

Important: Solutions have to be in **pdf** format. Every question must be on a **separate** pdf and submitted through Crowdmark. We prefer solutions prepared on LaTeX but that is **not required**. We will **not** accept jpg photos of handwritten solutions. We will not accept Word documents. Either of these formats can be exported to pdf. **TAs will remove marks for blurry/illegible submissions.**

Question 1. [20 points] Provide context-free grammars which generate each of the following languages and *explain* how your grammars work in <u>one to three sentences</u>. Proofs of correctness for your grammars are not required. **Marks will be removed if no explanation is given even if your grammar is correct.**

1. [5 points] $L_a = \{a^n b^m : n, m \in \mathbb{N}, n, m \geq 1, |n - m| \leq 1\}$
2. [5 points] $L_b = \{vwvw^R : v, w \in \{a, b\}^*, |v| = 2\}$ Note: w^R is the reverse of the string w
3. [5 points] $L_c = \{w \in \{a, b, c\}^* : n_a(w) + n_c(w) = n_b(w)\}$
4. [5 points] $L_d = \{a^n b^m c^k : n, m, k \in \mathbb{N}, m = n + 2k\}$

Question 2. [20 points] Let Σ be a (non-empty) alphabet and let $w \in \Sigma^*$ be a string. We say that $x \in \Sigma^*$ is a *prefix* of the string w if there exists a string $u \in \Sigma^*$ such that $w = xu$.
Consider the following language

$$L = \{w \in \{a, b\}^* : \text{for every prefix } x \text{ of } w \ n_b(x) \geq n_a(x)\}$$

Prove that L is a context-free language. Your proof should rely on *mathematical induction.*

Question 3. [20 points] Consider the grammar $G = (\{S, A\}, S, \{a, b\}, P)$ with the following set of production rules P

$$S \to aS | baA | abaA$$
$$A \to aAbb | aAb | \varepsilon$$

1. Determine what the language generated by G is. You may only use set builder notation and/or language operators in your answer. A proof is not required.
2. Show that the grammar G is ambiguous by
   (a) Providing two distinct leftmost derivations for the same string $w \in L(G)$.
   (b) Providing two distinct parse trees for the same string $w \in L(G)$.

Question 4. [20 points] State whether the following claim is true or false and prove your answer.

Claim: For any (non-empty) alphabet Σ, there is no language $L \subseteq \Sigma^*$ which is both regular *and* inherently ambiguous.

Question 5. [20 points] Provide PDA (either a NPDA or a DPDA) which accept the following languages and *explain* how your PDA work in one to three sentences. Proofs of correctness for your constructions are not required. **Marks will be removed if no explanation is given even if your PDA is correct.**

1. [10 points] $L_1 = \{w \in \{a, b\}^* : n_a(w) > 2 \cdot n_b(w)\}$
2. [10 points] $L_2 = \{a^n b^m c^k : n, m, k \in \mathbb{N}, n + k = m\}$

Supplementary Questions

These questions are not for credit. Do not submit them to `CrowdMark`. Nothing even remotely close to these questions will appear on the final, but they are quite fun so consider trying them in case you finish this assignment early.

Question 6. [0 points] Consider the alphabet $\Gamma = \{a, b, (,), +, \cdot, ^*, \emptyset, \epsilon\}$ where Γ can be thought of as the set containing all the symbols that can be used to form valid regular expressions over the alphabet $\Sigma = \{a, b\}$. Note that we use ϵ in Γ to disambiguate with the empty string ε. Provide a context-free grammar that generates exactly the set of valid regular expressions over the alphabet $\Sigma = \{a, b\}$. Prove the correctness of your grammar.

Question 7. [0 points] Given a grammar $G = (V, S, T, P)$, we say that G is a *linear* grammar (LG) if each of its production rules has the form $\alpha \to \beta$ where $\alpha \in V, \beta \in T^* \cdot (V \cup \{\varepsilon\}) \cdot T^*$. Suppose Σ is some (non-empty) alphabet and consider the family of languages generated by linear grammars

$$L_{\text{LG}} = \{L \subseteq \Sigma^* : \exists \text{ LG } G \text{ such that } L(G) = L\}$$

In addition, recall the family of languages generated by *right-linear grammars* (i.e. the regular languages) as well as the family of languages generated by *context-free grammars* (i.e. the context-free languages)

$$L_{\text{REG}} = \{L \subseteq \Sigma^* : \exists \text{ RLG } G \text{ such that } L(G) = L\}$$
$$L_{\text{CF}} = \{L \subseteq \Sigma^* : \exists \text{ CFG } G \text{ such that } L(G) = L\}$$

One of the following 4 claims is true. Which one is it? Prove your answer.

1. $L_{\text{REG}} \subset L_{\text{LG}} \subset L_{\text{CF}}$
2. $L_{\text{REG}} \subseteq L_{\text{LG}} \subset L_{\text{CF}}$
3. $L_{\text{REG}} \subset L_{\text{LG}} \subseteq L_{\text{CF}}$
4. $L_{\text{REG}} \subseteq L_{\text{LG}} \subseteq L_{\text{CF}}$

Question 8. [0 points] Let Σ be some alphabet where $|\Sigma| \geq 2$. We define the following language operator

$$\texttt{cycle}(L) = \{yx : \exists x, y \in \Sigma^* \text{ such that } xy \in L\}$$

Prove the following claims.

1. If L is regular, then so is $\texttt{cycle}(L)$.
2. If L is context-free, then so is $\texttt{cycle}(L)$.