CS 301: Languages and Automata **Fall 2015**

Problem Set #2

Due: Tuesday, October 20, 2015

Problem 1 [30 pts] Give a PDA that recognizes each of the following languages. The input alphabet is $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{\#}\}$. You may use whatever stack alphabet Γ you wish. Turn in your solutions to Blackboard as separate JFLAP files. Make sure you test your solutions! [*Hint: Parts b and c will require nondeterminism.*]

a. $\{x\mathtt{\#}y \mid x, y \in \{\mathtt{a}, \mathtt{b}\}^*$ and $x^{\mathcal{R}}$ is a substring of $y\}$. [*Hint:* $y = \{\mathtt{a}, \mathtt{b}\}^* x^{\mathcal{R}} \{\mathtt{a}, \mathtt{b}\}^*$]

b. $\{x\mathtt{\#}y \mid x, y \in \{\mathtt{a}, \mathtt{b}\}^*$ and $y^{\mathcal{R}}$ is a substring of $x\}$.

c. $\{x_1\mathtt{\#}x_2\mathtt{\#}\cdots\mathtt{\#}x_k \mid$ each $x_i \in \{\mathtt{a}, \mathtt{b}\}^*$ and some $x_i = x_j^{\mathcal{R}}\}$.

Problem 2 [10 pts] Prove that the class of context-free languages is closed under reversal. [*Hint: Consider a CFG in CNF and use induction on the length of the strings.*]

Problem 3 [10 pts] Prove that every string $w \neq \varepsilon$ generated by a CFG in CNF is derived in exactly $2n - 1$ steps. [*Hint: Use induction on the length of* w.]

Problem 4 [10 pts] Prove that the following CFG generates the language

$$\{xy \mid x, y \in \Sigma^*,\ |x| = |y|,\ \text{and } x \neq y\}.$$

$$\begin{aligned} S &\to AB \mid BA \\ A &\to XAX \mid \mathtt{a} \\ B &\to XBX \mid \mathtt{b} \\ X &\to \mathtt{a} \mid \mathtt{b} \end{aligned}$$

[*Hint: Consider* m *applications of the rule* $A \to XAX$ *and* n *applications of the rule* $B \to XBX$ *in the derivation of a string and note that each instance of an* X *gives you a single terminal. So*

$$S \Rightarrow AB \stackrel{*}{\Rightarrow} X^m \mathtt{a} X^m B \stackrel{*}{\Rightarrow} X^m \mathtt{a} X^m X^n \mathtt{b} X^n$$

where

$$X^i = \underbrace{XX\cdots X}_{i}.$$

All strings derived from the rightmost expression have length $2(m+n+1)$*. Now divide such a string into two* $m+n+1$ *parts and show that the two parts differ in at least one position. There's a similar argument when the first step in the derivation is* $S \Rightarrow BA$*.]*

Problem 5 [10 pts] We say that a CFG is in *forward-simple form* if all of its rules have the form $A \to tB$ or $A \to \varepsilon$ where A and B are variables (possibly the same variable), and t is a terminal. Prove that if G is a CFG in forward-simple form, then $L(G)$ is regular. [*Hint: Let* $G = (V, \Sigma, R, S)$ *be in FSF and build an NFA* $N = (Q, \Sigma, \delta, q_0, F)$ *that recognizes* $L(G)$*.*]

Problem 6 [10 pts] Call an alphabet *ordered* if it has an associated total order $\leqslant$.[1] For example, we can consider the alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ an ordered alphabet by defining $\mathtt{a} < \mathtt{b}$ and in fact, we can turn any alphabet into an ordered alphabet by defining a total order. For strings over an ordered alphabet, define an operation on strings $\textsc{Sort}(w) = w'$ such that the symbols of w' are the symbols of w in sorted order. For example, $\textsc{Sort}(\mathtt{aabab ab}) = \mathtt{aaaabbb}$. (Note, $\textsc{Sort}$ is *not* a homomorphism.)

For a language A defined over a sorted alphabet, define

$$\textsc{Sort}(A) = \{\textsc{Sort}(w) \mid w \in A\}.$$

For example, $\textsc{Sort}(\Sigma^*) = \mathtt{a}^*\mathtt{b}^*$, for $\Sigma = \{\mathtt{a}, \mathtt{b}\}$.

After all of that set up, here's the problem: Prove that if A is a regular language defined over the sorted alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$, then $\textsc{Sort}(A)$ is context-free. [*Hint: Given a DFA* $M = (Q, \Sigma, \delta, q_0, F)$ *that recognizes* A*, build a CFG that generates* $\textsc{Sort}(A)$*. Your CFG will have a form similar to, but different from forward-simple form (see Problem 5).*]

Problem 7 [Extra credit: 10 pts] Give a regular language A defined over the sorted alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ such that $\textsc{Sort}(A)$ is not context-free. Prove that $\textsc{Sort}(A)$ is not context-free. You may use the fact that $\{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n \mid n \geqslant 0\}$ is not context-free without proof.

[1] See `https://en.wikipedia.org/wiki/Total_order` for the definition of a total order.