CS 154 - Introduction to Automata and Complexity Theory

Spring Quarter, 2000

Assignment #3 - Due date: Wednesday, 4/26/00

Problem 1. *(10 points)* The *symmetric difference* of two languages L_1 and L_2 is the set of strings that are in exactly one of the two languages. Prove the following closure property: *If L_1 and L_2 are both regular, then so is the symmetric difference of L_1 and L_2.*

Problem 2. *(15 points)* The *derivative* of a language $L \subseteq \Sigma^*$ with respect to a symbol $a \in \Sigma$ is defined as:

$$\frac{\delta L}{\delta a} = \{w \in \Sigma^* \mid aw \in L\}.$$

The derivative contains all strings which can be obtained by taking a string in L that begins with an a and removing the a. (Strings in L that do not begin with an a are dropped completely.) For example, if $L = \{00, 110, 011, 001, 0000, 11111\}$, then we have $\frac{\delta L}{\delta 0} = \{0, 11, 01, 000\}$. Show that: if L is regular, then $\frac{\delta L}{\delta a}$ is regular for all $a \in \Sigma$.

(**Hint:** Consider a DFA for the language L and modify it to obtain a DFA for the language $\frac{\delta L}{\delta a}$.)

Problem 3. *(15 points)* You are told that the language $L_1 = \{0^{n^3} 1^n \mid n \geq 0\}$ is *not* regular. Now, prove that the language

$$L_2 = \{a^k\, b\, c^l \mid k, l \geq 0 \text{ and } k \neq l^3\}$$

is not regular *using the closure properties for regular languages.*

Problem 4. *(20 points)* In class we proved the following statement: *Given a DFA M with n states, the language $L(M)$ is infinite if and only if there exists a string $w \in L(M)$ such that $n \leq |w| < 2n$.* We then used this result to describe an algorithm for solving the following decision problem: *Given a regular language L, is it infinite.*

In this problem, you are required to provide an algorithm to solve the following decision problem: *Given a regular language L, does it contain at least 7 strings.* You must justify the correctness of your algorithm.

Problem 5. *(20 points)* Let G be the context-free grammar with the following productions.

$$S \rightarrow aS \mid Sb \mid a \mid b$$

a). [15 points] Prove that all strings $w \in L(G)$ have the property that they *do not* contain ba as a substring. We suggest a proof by induction on the length of the derivation.
b). [5 points] Describe the language $L(G)$. Justify your answer using part (a).

Problem 6. *(20 points)* This problem explores the significance of the notion of ambiguity of CFGs.

(a). Consider the grammar G with the productions

$$E \longrightarrow E+E \mid E*E \mid (E) \mid x \mid y$$

This grammar generates arithmetic expressions involving the variables x and y. This grammar is ambiguous for two reasons. First, the string $x+y+x$ has two derivation trees and leftmost derivations. Also, as we pointed out in the class, $x+y*x$ has two distinct leftmost derivations. Construct an unambiguous grammar G^* such that $L(G^*) = L(G)$. While we do not require a formal proof of correctness, you must provide a brief justification for your solution. (Note: The first kind of ambiguity does not have any serious repercussions since $(x+y)+x = x+(y+x)$. However, the higher precedence of multiplication implies that $x+(y*x)$ is the **right** interpretation of the second expression and your grammar should reflect this fact.)

(b). The following context-free grammar generates arithmetic expressions in the postfix notation which is used, for example, in the programming language APL. Note that the previous grammar was based on the infix notation. Is this grammar ambiguous? Give a brief justification.

$$S \longrightarrow SS+ \mid SS- \mid SS* \mid x \mid y$$

(**Hint:** To better understand the grammar, try to construct a derivation tree for the string $xxy*x-+$.)

Reading Assignment

We are done with Chapter 4 of the course reader. We will now start covering Chapters 5 and 6 from the reader – Section 5.4 will not be covered in detail but makes for very interesting background reading. (In the Hopcroft-Ullman book, these correspond to Chapters 4 and 5.)