CS 271: Automata and Computation Theory **Spring 2014**

Problem Set #1

Due: Thursday, February 13, 2014

Problem 1 Let $L = \{w \mid w \text{ contains at least two } \texttt{0} \text{ and at most one } \texttt{1}\}$. Construct an NFA that recognizes L^*.

Problem 2 Prove that every NFA can be converted to an equivalent one that has a single accept state.

Problem 3 Prove that regular languages are closed under complement. [*Hint: given a DFA* $M = (Q, \Sigma, \delta, q_0, F)$ *that recognizes a language* L*, build a new DFA* $M' = (Q', \Sigma, \delta', q_0', F')$ *that recognizes* $L^{\mathcal{C}} = \Sigma^* \setminus L = \{w \mid w \notin L\}$.]

Problem 4

a. Use the result from Problem 3 along with other closure properties of regular languages to show that regular languages are closed under set difference. That is, given regular languages L_1 and L_2, show that

$$L_1 \setminus L_2 = \{w \in L_1 \mid w \notin L_2\}$$

is regular.

b. Show that regular languages are closed under symmetric set difference

$$L_1 \triangle L_2 = \{w \mid \text{either } w \in L_1 \text{ or } w \in L_2 \text{ but not both}\}.$$

Problem 5

a. For any language L, we defined

$$\textsc{Prefix}(L) = \{w \mid \exists x \in \Sigma^* \text{ s.t. } wx \in L\}.$$

Prove that regular languages are closed under $\textsc{Prefix}$.

b. For any language L, we defined

$$\textsc{Suffix}(L) = \{w \mid \exists x \in \Sigma^* \text{ s.t. } xw \in L\}.$$

Using closure properties of regular languages and the result of part **a**, prove that regular languages are closed under $\textsc{Suffix}$.

Problem 6

a. Let Σ and Γ be alphabets and let $f : \Sigma \to \Gamma$ be a function that maps symbols in Σ to symbols in Γ. One such example is $f : \{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{4}, \mathtt{5}\} \to \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$ given by

$$\begin{aligned} f(\mathtt{1}) &= \mathtt{b} \\ f(\mathtt{2}) &= \mathtt{b} \\ f(\mathtt{3}) &= \mathtt{a} \\ f(\mathtt{4}) &= \mathtt{d} \\ f(\mathtt{5}) &= \mathtt{a}. \end{aligned}$$

We can extend such an f to operate on strings $w = w_1 w_2 \cdots w_n$ by $f(w) = f(w_1)f(w_2)\cdots f(w_n)$. Using the same example, $f(\mathtt{132254}) = \mathtt{babbad}$. We can extend f to operate on languages by $f(L) = \{f(w) \mid w \in L\}$.

Prove that if L is a regular language and $f : \Sigma \to \Gamma$ is an arbitrary function—that is, it is *not* necessarily the example given above—then $f(L)$ is regular. [*Hint: given a DFA M that recognizes L, build an NFA N that recognizes $f(L)$ by applying f to the symbols on each transitions.*]

b. A homomorphism is a function $f : \Sigma \to \Gamma^*$ that maps symbols in Σ to *strings* over Γ. One example of a homomorphism is the function that maps every string to ε. A less-trivial example is $f : \{\mathtt{a}, \mathtt{b}\} \to \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ given by

$$\begin{aligned} f(\mathtt{a}) &= \mathtt{bacca} \\ f(\mathtt{b}) &= \mathtt{b}. \end{aligned}$$

As before, we can extend f to operate on strings $w = w_1 w_2 \cdots w_n$ by $f(w) = f(w_1)f(w_2)\cdots f(w_n)$ and languages by $f(L) = \{f(w) \mid w \in L\}$.

Prove that regular languages are closed under homomorphism. [*Hint: As with your construction in part **a**, you want to apply f to the symbols on each transition but in this case you may need to add additional states if the length of $f(a)$ is not 1. Be sure to handle the case where $f(a) = \varepsilon$.*]

Problem 7 For languages L_1 and L_2, define

$$L_1 \obslash L_2 = \{w \in L_1 \mid w \text{ does not contain any string in } L_2 \text{ as a substring}\}.$$

Prove that regular languages are closed under $\obslash$.[1] [*Hint: Think about what $\Sigma^* \circ L \circ \Sigma^*$ means for a language L. Write $L_1 \obslash L_2$ in terms of set difference and concatenation and apply closure properties of regular languages.*]

Problem 8 For each language in Exercise 1.6 in Sipser, give an equivalent regular expression. (You don't need to prove that it's correct.)

[1] You can typeset $\obslash$ in LaTeX by putting the line `\usepackage{mathabx}` in the preamble and using `\obackslash` in math mode.

Problem 9 Using the procedure given in Lemma 1.55 in Sipser, convert the regular expression $(0 \cup 11)^*01(00 \cup 1)^*$ to an NFA. Show each step.

Problem 10 Using the procedure given in Lemma 1.60 in Sipser, convert the following DFA to a regular expression. Show each step.