# Introduction to the Theory of Computation
# Homework 2
# Due 9/15/2017
# These are problems

Saad Mneimneh
Computer Science
Hunter College of CUNY

**Problem 0: Readings**
(a) Read Chapter 0 in Sipser's book.

(b) Read Chapter 1 in Sipser's book.

**Problem 1: Reversing a regular language**
For any string $w = w_1 w_2 \ldots w_n$, the reverse of $w$, written $w^R$, is the string $w$ in reverse order, $w_n \ldots w_2 w_1$. For any language $A$, let $A^R = \{w^R | w \in A\}$. Show that if $A$ is regular, so is $A^R$.

**Problem 2: A binary adder**
Let

$$\sum = \left\{ \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \ldots, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}$$

$\Sigma$ contains all size 3 columns of 0s and 1s. A string of symbols in $\Sigma$ gives three rows of 0s and 1s. Consider each row to be a binary number and let

$$B = \{w \in \Sigma^* | \text{the bottom row of } w \text{ is the sum of the top two rows}\}$$

Show that $B$ is regular. *Hint*: it is easier to work with $B^R$ and construct a finite automaton that checks the sum from right to left. Assume the result obtained in Problem 1.

**Problem 3: Shift and add**
Let

$$\sum = \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\},$$

all columns of 0s and 1s of height two. A string of symbols in $\Sigma$ gives two rows of 0s and 1s. Consider each row to be a binary number and let

$$C = \{w \in \Sigma^* | \text{the bottom row of } w \text{ is three times the top row}\}$$

Show that $C$ is regular.

**Problem 4: Prefix-free language**
For a given language $A$, let

$$\text{NOPREFIX}(A) = \{w \in A | \text{ no proper prefix of } w \text{ is a member of } A\}$$

Show that the class of regular languages is closed under the NOPREFIX operation.

**Problem 5: Drop half**
If $D$ is any language, let $D_{\frac{1}{2}-}$ be the set of all first halves of strings in $D$, so

$$D_{\frac{1}{2}-} = \{x | \text{ for some } y, |x| = |y| \text{ and } xy \in D\}$$

Show that if $D$ is regular, then so is $D_{\frac{1}{2}-}$. *Hint*: given DFA $M$ such that $L(M) = D$, construct an NFA that moves from the start state following valid transitions of $M$, and in parallel from some accept state of $M$ by guessing backwards the string $y$. Give the formal definition of such an NFA.

**Problem 6: Pumpable non-regular**
Let $L \subset \Sigma^*$ be a language that is not regular, and let $a \notin \Sigma$.

(a) Show that the language $\{aw | w \in L\}$ is not regular. *Hint*: do not use the pumping lemma because you don't know what $L$ looks like.

(b) Consider the language $L'$:

$$L' = \{aw | w \in L\} \cup \{a^k \Sigma^* | k \neq 1\}$$

Show that $L'$ is not regular. *Hint*: consider the intersection of $L'$ and the language generated by the regular expression $a\Sigma^*$, and use of a closure property for regular languages.

(c) Show that every string in $L'$ can be pumped.

**Problem 7: Equivalence relation of DFA (alternative to pumping)**
Given a DFA $M = (Q, \Sigma, \delta, q_0, F)$, define $M(w) = q = \delta(\ldots\delta(\delta(q_0, w_1), w_2)\ldots, w_n)$, i.e. following the transitions based on the symbols of $w$, puts $M$ in state $q$.

Define $w_1 \sim_M w_2$ to mean that $M(w_1) = M(w_2)$. Also define the extension of $w$ with respect to a language $L$ as:

$$E_L(w) = \{z | wz \in L\}$$

We say that $w_1 \equiv_L w_2$ iff $E_L(w_1) = E_L(w_2)$. Observe that $\equiv_L$ is an equivalence relation.

(a) Find the equivalence classes of the following languages (you can list the different possible $E_L(.)$):

- $\{ab, ac, bb, bc\}$
- $a^*b^*$
- $\{a^n b^n | n \geq 0\}$

(b) Show that if $w_1 \sim_M w_2$, then $w_1 \equiv_{L(M)} w_2$.

(c) Show that if $L$ is regular, then $M$ has to have at least as many states as the number of equivalence classes of the relation $\equiv_{L(M)}$, and conclude that if the number of equivalence classes of language $L$ is infinite, then $L$ is not regular.