CS 5000: Theory of Computability
Assignment 2

Vladimir Kulyukin
Department of Computer Science
Utah State University

September 8, 2017

Learning Objectives

1. Closure Properties of Regular Languages
2. Programmatic Implementation of DFAs
3. Equivalence of NFAs and DFAs

Problem 1: Closure Properties of Regular Languages (2 points)

Let L_1 and L_2 be two languages. Recall that the concatenation of L_1 and L_2 is defined as $L_1 \times L_2 = \{xy | x \in L_1 \wedge y \in L_2\}$.

The Kleene closure of a language L is $L^* = \{x_1 x_2 x_3 ... x_n | n \geq 0, x_i \in L, 0 \leq i \leq n\}$.

The difference between L_1 and L_2 is $L_1 - L_2 = \{x | x \in L_1 \wedge \neg(x \in L_2)\}$.

The quotient of L_1 and L_2 is defined as $L_1/L_2 = \{x | \exists y \in L_2 \wedge xy \in L_1\}$.

Prove the following statements:

1. Show that regular languages are closed under concatentation.
2. Show that regular languages are closed under the Kleene closure.
3. Show that regular languages are closed under difference.
4. Show that regular languages are closed under quotients with final languages, i.e., languages that consist of finite numbers of strings.

Problem 2: (1 point)

Consider that following NFA $M = (Q, \Sigma, \delta, q_0, F)$, where $Q = \{q_0, q_1, q_2\}$, $\Sigma = \{0, 1\}$, $F = \{q_1\}$, and the transition function δ is defined as follows:

1. $\delta(q_0, 0) = \{q_0, q_1\}$.
2. $\delta(q_0, 1) = \{q_1\}$.
3. $\delta(q_1, 0) = \{q_2\}$.
4. $\delta(q_1, 1) = \{q_2\}$.
5. $\delta(q_2, 1) = \{q_2\}$.

Convert M into an equivalent dfa and draw it.

Problem 3: Programmatic Implementation of Finite State Machines (2 points)

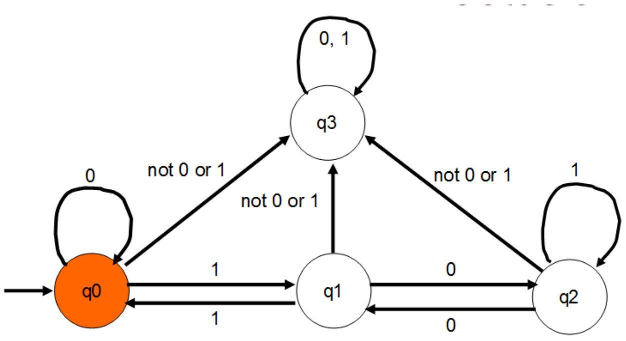

Figure 1: Module 3 DFA.

Fig. 1 shows a DFA that processes strings of 0s and 1s that are binary representations of numbers divisible by 3. Implement this DFA in Java or Python. Define a class Mod3DFA with a public method $processString(Stringx)$ so that $Mod3DFA.processString(Stringx)$ returns true if $x$is the binary representation of a number divisible by 3 and false otherwise. Save your solution in $Mod3DFA.java$ or $Mod3DFA.py$ and submit it via Canvas along with your solutions to Problems 1 and 2.

What to Submit

Save your solutions in hw02.zip and submit it via Canvas.