HW Solution

CS/ECE 374: Algorithms & Models of Computation, Spring 2019 Version: **1.0**

Submitted by:

- ≪**Alan Lee**≫: ≪**alanlee2**≫
- ≪**Joshua Burke**≫: ≪**joshuab3**≫
- ≪**Gerald Kozel**≫: ≪**gjkozel2**≫

(100 PTS.) Then, shalt thou find two runs of three.
Let L be the set of all strings in $\{0,1\}^*$ that contain the substrings 000 and 111.

A (60 PTS.) Describe a DFA that over the alphabet $\Sigma = \{0,1\}$ that accepts the language L. Argue that your machine accepts every string in L and nothing else, by explaining what each state in your DFA *means*.

You may either draw the DFA or describe it formally, but the states Q, the start state s, the accepting states A, and the transition function δ must be clearly specified.

B (40 PTS.) Give a regular expression for L, and briefly argue why the expression is correct.

5 Solution:

A (See attached sheet for DFA.)

- q0 is the starting state.
- q1, q2, q3 are states for successive runs of 0; q6, q7, q8 are that for runs of 1. If a run is interrupted on either side, there is a transition back to either q1 or q6, depending on what run was interrupted. (q1 for a sudden 0, q6 for a sudden 1.)
- q3 is a state representing the case when 000 has already occurred, but 111 hasn't.
- q8 is similar to q3, but the opposite (111 has occurred but 000 hasn't).
- States q4 and q5 follow a run of consecutive 1's. States q9 and q10 are that for 0's.
- State q11 is the accepting state, and any number of 0's or 1's after reaching this state are acceptable.

B $((0+1)^*(000)(0+1)^*(111)(0+1)^*) + ((0+1)^*(111)(0+1)^*(111)(0+1)^*)$

This expression is correct because the criteria which it must satisfy is that it must contain both sub-strings 000, and 111. The above expression satisfies both cases of either order of 000 111 or 111 000 with any combination of 0's an 1's in between.