CS500, Theory of Computation
Homework #1

Note: You may discuss this homework with others in the class. However, you must do your own writeup, and you must clearly state on your homework who you worked with. Due by email in .pdf format by midnight on Wednesday, February 4th.

1. (Exercise 1.4 c, d, and e, and Exercise 1.13) Draw diagrams of DFAs that recognize the following languages, and give regular expressions for each one. In all cases the input alphabet is $\Sigma = \{0, 1\}$.

   (a) $\{w \mid w \text{ contains the substring } 0101\}$

   (b) $\{w \mid w \text{ has length at least 3 and its third symbol is } 0\}$

   (c) $\{w \mid w \text{ starts with 0 and has odd length, or starts with 1 and has even length}\}$

2. Recall the definition $\delta^*(q, w_1 w_2 \cdots w_n) = \delta^*(\delta(q, w_1), w_2 \cdots w_n)$ for the effect on a DFA of reading a word w. The language L recognized by a machine can then be written

   $$L = \{w \in \Sigma^* \mid \delta^*(q_0, w) \in F\}$$

   where q_0 is the initial state and F is the set of accepting states.

   Given a DFA M, we can define an equivalence $\sim_M$ on words $u, v \in \Sigma^*$ as follows:

   $$u \sim_M v \text{ if and only if } \delta^*(q_0, u) = \delta^*(q_0, v)$$

   and, as we discussed in class, given a language L we can define an equivalence $\sim_L$ as

   $$u \sim_L v \text{ if and only if } (\forall w : uw \in L \Leftrightarrow vw \in L)$$

   Prove formally that if M recognizes L, then $u \sim_M v$ implies $u \sim_L v$. Explain why the converse might not be true—that is, why $\sim_M$ might be a "finer" equivalence, dividing Σ^* into more equivalence classes, than $\sim_L$.

3. Using the same notation as in the previous problem, prove that if $u \sim_L v$, then for any $a \in \Sigma$ we have $ua \sim_L va$. Show that this allows us to can define a DFA M whose set of states Q is the set of equivalence classes of $\sim_L$; define q_0, δ and F, and prove that M recognizes L. Moreover, prove that M is the smallest DFA that recognizes L, and that it is (up to isomorphism) the only DFA of this size that recognizes L.

4. What are the equivalence classes of $\sim_L$ if $L = \{w \in \{0, 1\}^* | w \text{ contains the substring } 101\}$? Use this to draw the minimal DFA for L.

5. (Exercise 1.17c, modified) Give two proofs, one based on the *pumping lemma* (which will will talk about this week) and one based on equivalence classes, that the language $\{a^{2^n} \mid n \in \mathbb{N}\} = \{a, aa, aaaa, aaaaaaaa, \ldots\}$ is not regular.

6. (Problem 1.30) Show that for all p, the language

$$L_p = \{w \in \{0,1\}^* \mid w \text{ represents a binary integer divisible by } p\}$$

is regular.

7. (Problem 1.39) Prove that for all $k > 1$, DFAs with k states are more powerful than DFAs with $k-1$ states. That is, come up with a family of languages L_k such that L_k can be recognized by a DFA with k states but not by a DFA with $k-1$ states. Thus the DFAs form a strict hierarchy based on the number of states, which classify the regular languages according to their complexity.

8. (Problem 1.44 — a little harder) Now prove that NFAs can be exponentially more compact than DFAs, i.e., that the exponential blowup in the number of states we get when converting an NFA to a DFA is sometimes necessary. That is, give a family of languages L_k such that L_k can be recognized by an NFA with k states, but whose minimal DFA has $\Omega(2^k)$ states. Hint: Look at Example 1.14 in Sipser.

9. (Problem 1.24 plus...) Given a word $w = w_1 \cdots w_n$, its *reverse* is $w^R = w_n \cdots w_1$, and the reverse of a language L is $L^R = \{w^R \mid w \in L\}$. Show that if L is regular, so is L^R. However, show also that the number of states for the minimal DFA for L^R can be exponentially larger than the number for L. Hint: use the same languages as in the previous problem.

10. (Problem 1.41) As we showed in class, the language

$$L = \{w \mid w \text{ has an equal number of 0s and 1s}\}$$

is not regular. However, prove that the language

$$L = \{w \mid w \text{ has an equal number of 01s and 10s}\}$$

is regular.

11. (Problem 1.42 — Tricky!) For a language L, define $L_{1/2}$ as

$$L_{1/2} = \{x \mid xy \in L \text{ for some word } y \text{ such that } |y| = |x|\}$$

That is, $L_{1/2}$ is the set of "first halves" of L, namely the set of words x that can be followed by words y of the same length giving a word xy in L. Prove that if L is regular, so is $L_{1/2}$.

12. Show that

$$L = \{w \in \{0,1\}^* \mid w \text{ represents a prime number in binary}\}$$

is not regular.