CSE 355 HOMEWORK ONE
DUE 16 SEPTEMBER 2015, START OF CLASS

(1) Give a state diagram and the table specifying the transition function, for a DFA to recognize
(a) \( L_1 = \{ w \in \{0,1\}^* : w \text{ either starts with } 10 \text{ or ends with } 01, \text{ but not both} \} \)
(b) \( L_2 = \{ w \in \{0,1\}^* : w \text{ contains } 00 \text{ as a substring an odd number of times} \} \) Note: 000 contains two occurrences of 00 as a substring.
(c) \( L_3 = \{ w \in \{0,1,2\}^* : w \text{ is the ternary (base 3) representation of a number that is divisible by 2} \} \)
(d) (not to be graded) \( L_4 = \{ w \in \{0,1,2,3\}^* : w \text{ is the quaternary (base 4) representation of a number that is divisible by 4} \} \)
(e) (not to be graded) \( L_5 = \{ w \in \{0,1\}^* : w \text{ is the binary representation of a number that is not divisible by 3 or 7} \} \)

(2) Give a state diagram for an NFA with as few states as you can to recognize
(a) \( L_6 = \{ w \in \{0,1\}^* : w \text{ starts with a 1 or ends with a 1} \} \)
(b) \( L_7 = \{ w \in \{0,1\}^* : w \text{ contains the substring } 010101 \} \)
(c) \( L_8 = L_6 \)
(d) (not to be graded) \( L_9 = \{ w \in \{0,1\}^* : w \text{ has odd length and an odd number of 1s, or starts with a 0} \} \)
(e) (not to be graded) \( L_{10} = \{ w \in \{0,1\}^* : w \text{ has more occurrences of the substring } 01 \text{ than of the substring } 10 \} \)

(3) Using the languages from Question 1 and the method of Theorem 1.25, give a state diagram and/or the table specifying the transition function, for a DFA to recognize
(a) \( L_1 \cap L_2 \)
(b) \( L_1 \cup L_3 \)
(c) \( L_1 \cap L_3 \)
(d) \( L_3 \cup L_3 \)
Do not simplify the DFA produced.

(4) If \( w = w_1 \cdots w_k \) \((w_i \in \Sigma \text{ for } 1 \leq i \leq k)\) is a string in \( \Sigma^* \), the string \( w^{\text{rev}} \), the reversal of \( w \), is the string \( w_k \cdots w_1 \). The reversal of language \( L \) is \( L^{\text{rev}} = \{ w^{\text{rev}} : w \in L \} \). I think that whenever \( L \) is regular, \( L^{\text{rev}} \) is also regular. My friend tells me that showing this is easy: Just take a DFA \( M \) that recognizes \( L \) and reverse all of the transitions to get a DFA that recognizes \( L^{\text{rev}} \). Does my friend’s idea work? Answer yes or no, and explain.

(5) A subsequence of a string \( w \) is defined to be a string obtained 0 or more symbols (not necessarily consecutive) from \( w \). For example, copter is a subsequence of computers. You are given a DFA \( M \) to recognize a regular language \( L \). You want to make an NFA that recognizes all subsequences of strings in \( L \), that is \( \text{subseq}(L) = \{ w \in \Sigma^* : w \text{ is a subsequence of } x \text{ for some } x \in L \} \). Devise (and explain) a general method for producing an NFA \( M' \) that recognizes \( \text{subseq}(L) \), given only the description of \( M \). Justify why your method works.