Theory of Computation

Fall 2025, Midterm Exam. (Solutions)

1. (20 pts) True or False? No penalty for wrong answers. No explanations are needed.

   (1) $|2^{\emptyset}| = 0$.
   (Sol.:) False. $|2^{\emptyset}| = |\{\emptyset\}| = 1$.

   (2) For any two languages L_1 and L_2, $(L_1)^* \cdot (L_2)^* \subseteq (L_1 \cup L_2)^*$.
   (Sol.:) True.

   (3) If L is regular and L' is not regular, then the language $L \cdot L'$ is always not regular.
   (Sol.:) False.

   (4) There are countably many context-free languages.
   (Sol.:) True.

   (5) The language $\{a^i b^j c^k \mid j = 2i + k\}$ is not a context-free language.
   (Sol.:) False. Can be accepted by a PDA.

   (6) If L^* is regular, then L must be regular as well.
   (Sol.:) False. Any language L containing 0 and 1 must have $L^* = \{0,1\}^*$.

   (7) The context-free grammar $S \rightarrow\ a \ \mid\ ab\ \mid SS\ \mid\ Sb$ is ambiguous.
   (Sol.:) True. ab has two parse trees.

   (8) If $M = (Q, \Sigma, \delta, q_0, F)$ is a minimal DFA accepting language L, then $\overline{M} = (Q, \Sigma, \delta, q_0, Q - F)$ is a minimal DFA accepting $\overline{L}$, i.e., the complement of L.
   (Sol.:) True.

   (9) Let $d(w)$ denote $|n - m|$ in which n is the number of 0s in w and m is the number of 1s in w. (For example, $d(00101) = 1$, as there are 3 0s and 2 1s.) The language $\{w \in \{0,1\}^* \ \mid d(w) = 0 \ mod\ 3\}$ is regular. (Note: $d(w) = 0\ mod\ 3$ iff $d(w)$ can be divided by 3.)
   (Sol.:) True.

   (10) Let M be a DFA with n states. If M accepts a string of length greater than n, then $L(M)$ must be infinite.
   (Sol.:) True.

2. (20 pts) Let M_1 and M_2 be the two finite automata over the alphabet $\{0,1\}$ shown in Figure-1 below. Let $L_1 = L(M_1)$ and $L_2 = L(M_2)$.

   (a) (7 pts) Using the product construction, construct a finite automaton M from M_1 and M_2, such that $L(M) = L_2 - L_1 = \{w \mid w \in L_2, w \notin L_1\}$. Indicate M's initial state, set of accepting states and fill-in the transition table in Figure-2 below. You may think of state RX as the pair (R, X). No explanations are needed.

   (b) (8 pts) Apply the minimization algorithm on the resulting automaton M. Fill-in the table in Figure-2 below by placing an "X" at the $i - j$ entry if states i and j cannot be merged. No explanations are needed.

   (c) (5 pts) Draw the minimal DFA from the table constructed above.

(Sol.:) Initial state is RX, set of accepting states $= SX, SY$

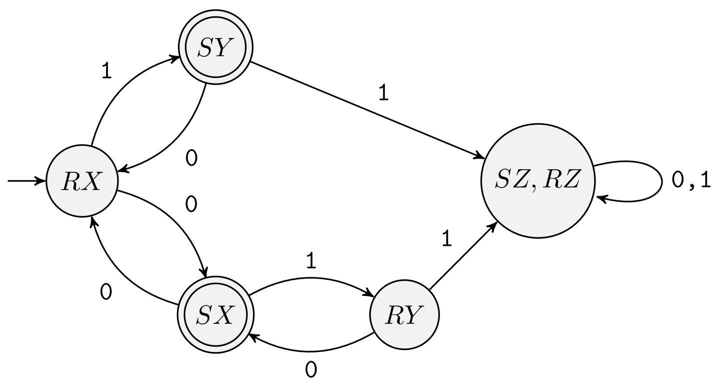
	0	1
RX	SX	SY
SX	RX	RY
RY	SX	SZ
SY	RX	RZ
RZ	SZ	SZ
SZ	RZ	RZ

Figure-2

	RX	SX	RY	SY	RZ
SX	X				
RY	X	X			
SY	X	X	X		
RZ	X	X	X	X	
SZ	X	X	X	X	A

Figure-3

(c)



3. (10 pts) The *Thue-Morse sequence* is an infinite sequence $w_1, w_2, w_3, ...$ over the alphabet $\{0, 1\}$, recursively defined as follows: $w_1 = 0$, $w_{i+1} = w_i\overline{w_i}$ for all $i \geq 1$, where $\overline{w}$ denotes the bitwise complementary string (e.g., $\overline{0110} = 1001$). Note that the first four strings are $w_1 = 0$, $w_2 = w_1\overline{w_1} = 01$, $w_3 = w_2\overline{w_2} = 0110$, $w_4 = w_3\overline{w_3} = 01101001$. Let $L = \{w_1, w_2, w_3, w_4, ...\}$.

   (a) (5 pts) Use the pumping lemma to show that L is not a regular language.
   (Hint: Note that all strings in L satisfy that $|w_{i+1}| = 2|w_i|$.)
   (Sol.:) Suppose L is regular. Let $p > 0$ be the pumping length. Choose $s = w_{p+1} = w_p\overline{w_p}$ and $s = xyz$ is the partition in pumping lemma. Obviously,we have$|w_p| < |xy^2z| < |w_{p+1}|$ which gives $xy^2z \notin L$.

   (b) (5 pts) Use the Myhill-Nerode theorem to show that L is not a regular language, i.e., show that the equivalence relation $x \equiv_L y$ has an infinite number of equivalence classes. (Recall that $x \equiv_L y$ iff $\forall z \in \Sigma^*(xz \in L \Leftrightarrow yz \in L)$.)
   (Sol.:) First, $\overline{w_i} \in L^{w_i}$ and $|\overline{w_i}| = 2^{i-1}$, L^{w_i} is not empty $\forall i$. And $\forall w \in L^{w_i}$, $|w| = 2^k - 2^{i-1}$ for some k, it is clear (converting the number to binary) that $\overline{w_i} \notin L^{w_j}$ if $i \neq j$. Thus all L^{w_i} form different equivalence classes.

4. (10 pts) A permutation of a string x is any string that can be obtained by rearranging the characters of x. Thus, for example, the string abc has exactly six permutations: $abc, acb, bac, bca, cab, cba$. For a language L over alphabet Σ, define

$$\text{SELECT}(L) = \{x \in \Sigma^* \mid \text{ every permutation of } x \text{ is in } L\}.$$

For example, if $L = \{ab, ba, ac, abc, bac, cba\}$, $\text{SELECT}(L) = \{ab, ba\}$. Answer the following questions:

   (a) (5 pts) Suppose $L = 1^*0$, what is SELECT(L)?
   (Sol.:) $\{0\}$

(b) (5 pts) Prove that regular languages are NOT closed under the operation SELECT.
(Hint: Consider the language $L = \overline{(01)^*}$, i.e., the complement of $(01)^*$. What is SELECT(L)?)
(Sol.:) SELECT(L) is the set of strings with an unequal number of 0's and 1's.

5. (10 pts) Given a language L over $\Sigma = \{a, b\}$, consider $Init(L) = \{x \in \Sigma^* \mid \text{ for some } y \in \Sigma^*, xy \in L\}$. We define homomorphism h and g from $\{a, b\}^*$ to $\{a, \hat{a}, b, \hat{b}\}^*$ as follows:

$$h(a) = a,\ h(b) = b,\ h(\hat{a}) = a,\ h(\hat{b}) = b,\ g(a) = a,\ g(b) = b,\ g(\hat{a}) = \varepsilon,\ g(\hat{b}) = \varepsilon$$

(a) (5 pts) What is $h^{-1}(\{aba\})$?
(Sol.:)$\{aba, \hat{a}ba, ab\hat{a}, \hat{a}b\hat{a}, a\hat{b}a, \hat{a}\hat{b}a, a\hat{b}\hat{a}, \hat{a}\hat{b}\hat{a}\}$

(b) (5 pts) Express $Init(L)$ using (some of) L, h, h^{-1}, g, g^{-1}, some regular expression R, and language operators such as $\cup, \cap, \cdot$. Briefly explain why your expression works. Be sure to write the regular expression R clearly.
(Sol.:) $Init(L) = g(h^{-1}(L) \cap (a + b)^*(\hat{a} + \hat{b})^*)$

6. (10 pts) Let L be a regular language accepted by a DFA $M = (Q, \Sigma, \delta, q_0, F)$. Construct an NFA $M' = (Q', \Sigma, \delta', q_0', F')$ to accept the language half(L) $= \{w \mid ww \in L\}$. Give the description of M' in detail, and briefly explain why your construction works.
(Sol.:)

$$\begin{aligned}
Q' &= Q \times Q \times Q \cup \{q_{start}\} \\
q_0' &= q_{start} \\
F' &= \{(q, q) | q \in Q\} \times F \\
\delta'((q_s, q_1, q_2), a) &= \{(q_s, \delta(q_1, a), \delta(q_2, a))\} \\
\delta'(q_{start}, \varepsilon) &= \{(q, q_0, q) | q \in Q\}
\end{aligned}$$

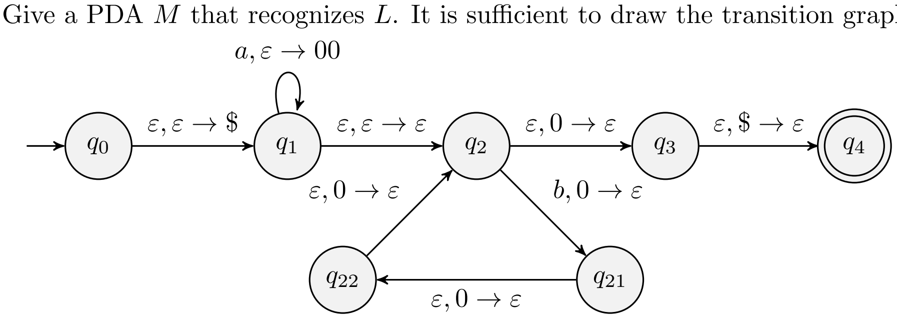
For state(q_s, q_1, q_2), q_s record the start state of path 2, and $q_1(resp. q_2)$ represent the state of path 1(resp.2).
$w \in L(M') \iff \delta'(q_{start}, w) \cap F' \neq \emptyset \iff \exists q_s \in Q$ s.t. $\delta(q_0, w) = q_s$ and $\delta(q_s, w) \in F \iff ww \in L$

7. (10 pts) Write a context-free grammar with three variables S, A, B to generate the language $\{0^i 1 0^j 1 0^k \mid i + k = 2j\}$. Briefly explain why your grammar works.
(Hint: note that i and k are both odd or even.)
(Sol.:) $S \to AB \mid 0A0B0;\quad A \to 00A0 \mid 1;\quad B \to 0B00 \mid 1$

8. (10 pts) Suppose we have the following language over the alphabet $\Sigma = \{a, b\}$

$$L = \{a^m b^n \mid 2m = 3n + 1, m, n \geq 0\}$$

(a) (6 pts) Give a PDA M that recognizes L. It is sufficient to draw the transition graph of M.



(Sol.:)

(b) (4 pts) Explain in Chinese or English in no more than 10 lines how M recognizes L.
(Sol.:) q0 to q1 pushes a bottom-of-stack marker ($) to start.
Loops at q1. For each a read, push two 0s onto the stack.
Cycles at q2 series, after m a's read. For each b read, pop 3 0s from the stack.
q2 to q3 pop one 0, pops $3m + 1$ 0's totally.
q3 to q4 remove the bottom-of-stack marker from stack.