Theory of Computation

Spring 2024, Midterm Exam. (April 9, 2024)

Solutions

---

(1) (10 pts) Answer the following questions:

(a) (5 pts) Give a language that cannot be recognized by a DFA with a single final state. Why?
Sol: $L = \{\epsilon, a\}$. For a DFA to accept ϵ, the initial state must be a final state. We need another state to accept a.

(b) (5 pts) Prove/disprove that any regular language can be recognized by an NFA with a single final state.
Sol: YES. For each final state of an NFA, add an ϵ transition to a new final state. Then then original final state can be made non-final.

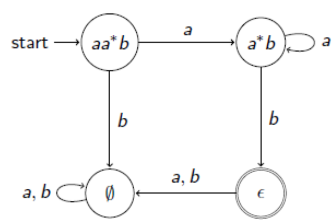
(2) (15 pts) Given a language $L \subseteq \Sigma^*$ and a $w \in \Sigma^*$, the *residual* of L with respect to w, denoted as L^w, is $L^w = \{u \in \Sigma^* \mid wu \in L\}$.

(a) (3 pts) State the *Myhill-Nerode Theorem* in terms of residuals of a language.
Sol: A language L is regular if and only if it has finitely many residuals.

(b) (8 pts) Consider language $L = aa^*b$. Complete the following table:

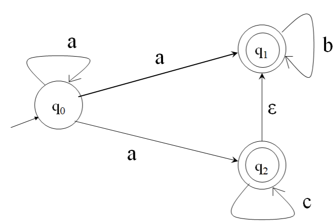
String w	Residual L^w
a	a^*b
b	$\emptyset$
ab	ϵ
ϵ	aa^*b

(c) (4 pts) Based on the results above, draw a 4-state DFA M accepting L. Note that each node in M should be labeled with a residual. Be sure to clearly mark the initial and the final states.



(3) (10 pts)

(a) (5 pts) Convert the following NFA M (over alphabet $\{a, b, c\}$) to an equivalent DFA using the subset construction

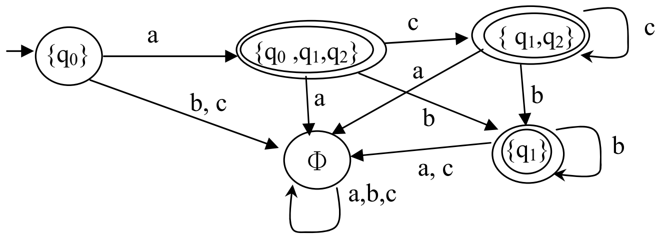


(b) (5 pts) Suppose we define a so-called *universal NFA* which is an NFA $M = (Q, \Sigma, \delta, q_0, F)$ except that the language it accepts is defined as $\{w \in \Sigma^* \mid \delta^*(q_0, w) \subseteq F\}$, i.e., to accept w all the computations must enter some states in F. Recall that the the original definition of acceptance for NFA is $\{w \in \Sigma^* \mid \delta^*(q_0, w) \in F\}$. Here $\delta^*(q_0, w)$ denotes the set of states reachable form q_0 upon reading w.

- (**Question**) Explain how to convert the DFA constructed in (a) to accept the language of M, assuming that M is a universal NFA.

(a) **Sol:**

(b)**Sol:** Make $\{q_0, q_1, q_2\}$ a non-final state, as q_0 is not a final state in the NFA.



(4) (5 pts) Given a language L, $perm(L)$ consists of all permutations of strings in L. For example, for $L = \{abc\}$, $perm(L) = \{abc, acb, bac, bca, cab, cba\}$. Suppose L is regular, is $perm(L)$ always regular? Justify your answer.
Sol: Not necessarily regular. Consider $L = (ab)^*$. $perm(L) = \{w \in \{a, b\}^* \mid \text{ the number of } a's = \text{ the number of } b's\}$, which is not regular.

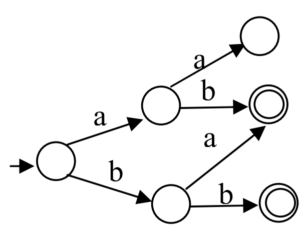
(5) (5 pts) Let L be a regular language. Is $\{w \in \Sigma^* \mid ww \in L\}$ always regular? Give a brief yet convincing argument.
Sol: YES. Suppose $M = (q, \Sigma, \delta, q_0, F)$ is a FA accepting L. Construct an NFA with initial state q_0' and each of its remaining states of the form (p, q, r) such that

- $q_0' \overset{\epsilon}{\rightarrow} (q, q_0, q)$, $\forall q \in Q$
- $(q, p, r) \overset{a}{\rightarrow} (q, p', r')$ if $p \overset{a}{\rightarrow} p'$ and $r \overset{a}{\rightarrow} r'$,
- (q, q, q_f) is a final state if $q_f \in F$.

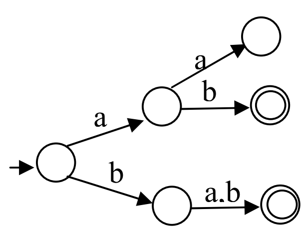
Note: the the construction is very similar to what we did for accepting $\frac{1}{2}L$ discussed in class.

(6) (10 pts) A DFA $M = (Q, \Sigma, \delta, q_0, F)$ is *reversible* if no input symbol can enter a state from two distinct states, i.e. for every $p, q \in Q$ and $a \in \Sigma$, if $\delta(p, a) = \delta(q, a)$, then $p = q$.

(a) (5 pts) Give a reversible DFA recognizing $L = \{ab, ba, bb\}$. Note that for a DFA, every input symbol must be defined on every state except for those sink states (i.e., states without outgoing transitions).



(b) (5 pts) Is there a unique minimal reversible DFA recognizing L (up to isomorphism)? Justify your answer.
Sol: No.



(7) (5 pts) Suppose we use the pumping lemma to show that

$$L = \{0^n 1^m \mid n, m \geq 1, m \text{ leves a remainder of 3 when divided by } n\}$$

is not regular. (For example, $0^4 1^7, 0^5 1^{13} \in L$.)
(Proof) Let p be the pumping constant. Consider $w = 0^{p+4} 1^{p+7}$. Let $w = xyz$ be any partition that $|xy| \leq p$, and $|y| > 0$. Suppose $x = 0^r$, $y = 0^s$, $z = 0^t 1^{p+7}$, where $r + s + t = p + 4$, $r + s \leq p$, $s > 0$. Now consider $xy^2z = 0^{r+2s+t} 1^{p+7} = 0^{p+4+s} 1^{p+7}$.

(**Question:**) Complete the proof by showing the remaining details that xy^2z is not in L.
Sol: Consider the following two cases:

- if $p + 4 + s \leq p + 7$, then $(p + 7 \; mod \; p + 4 + s) \leq 2$
- if $p + 4 + s > p + 7$, then $(p + 7 \; mod \; p + 4 + s) = p + 7 > 3$.

In either case, we have a contradiction.

(8) (20 pts) For each of the languages below, decide whether the language is context-free or not. Mark ◯ if it is context-free. mark × if not. No explanations are needed. No penalty for wrong answer.

(1) $\{0^n 1^n 1^n \mid n \geq 0\}$
Sol: ◯. The language is $\{0^n 1^{2n} \mid n \geq 0\}$

(2) $\{0^n 1^n 1^n 0^n \mid n \geq 0\}$
Sol: ×. The language is $\{0^n 1^{2n} 0^n \mid n \geq 0\}$

(3) $\{0^n 1^m \mid m, n \text{ are either both odd or both even}\}$
Sol: ◯. Use states to keep track of whether m, n is odd or even.

(4) $\{0^n 1^n 0^m 1^m \mid m, n \geq 0\}$
Sol: ◯ Can be accepted by a PDA.

(5) $\{(0^n 1^n)^m \mid m, n \geq 0\}$
Sol: ×. Use pumping lemma.

(6) $\{wxw \mid w, x \in \{0, 1\}^*\}$
Sol: ◯. Set w to ϵ. The language $= \{0, 1\}^*$.

(7) $\overline{\{a^n b^n c^n \mid n \geq 0\}}$, i.e., the complement of $\{a^n b^n c^n \mid n \geq 0\}$.
Sol: ◯.
The set is the union of (1) not of the form $a^*b^*c^*$; (2) $\{a^i b^j c^k \mid i \neq j\}$; (3) $\{a^i b^j c^k \mid i \neq k\}$; (4) $\{a^i b^j c^k \mid k \neq j\}$

(8) $\{a^i b^j \mid i + j = n^2, \text{ for some } n \geq 0\}$
Sol: ×. Intersecting with a^* yields $\{a^{n^2} \mid n \geq 0\}$, which is not CF.

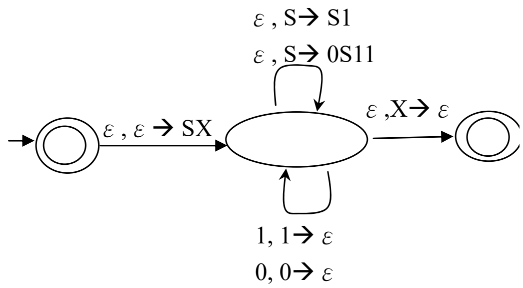
(9) $\{a^p b^q c^r \mid p, q, r \geq 0, \; p = r = 2q\}$
Sol: ×. Use pumping lemma.

(10) $\{a^p b^q c^r \mid p, q, r \geq 0, \; p + r = 2q\}$
Sol:◯. Can be accepted by a PDA.

(9) (10 pts) Consider language $L = \{0^i 1^j \mid 0 \leq 2i \leq j\}$.

(a) (5 pts) Give a context-free grammar for L.
Sol: $S \to 0S11 \mid S1 \mid \epsilon$

(b) (5 pts) Give a PDA for L. Draw the diagram of the PDA.



(10) (10 pts) Consider the following CFG G in CNF.

$$S \to AB \mid BC$$
$$A \to BA \mid a$$
$$B \to CC \mid b$$
$$C \to AB \mid a$$

Use CYK algorithm to decide whether string *baaba* is in $L(G)$ or not. Complete the following Table.

$X_{1,5}$				
$X_{1,4}$	$X_{2,5}$			
$X_{1,3}$	$X_{2,4}$	$X_{3,5}$		
$X_{1,2}$	$X_{2,3}$	$X_{3,4}$	$X_{4,5}$	
$X_{1,1}$ {B}	$X_{2,2}$ {A, C}	$X_{3,3}$ {A, C}	$X_{4,4}$ {B}	$X_{5,5}$ {A, C}
b	**a**	**a**	**b**	**a**

Recall that $X_{i,j} = \{A \mid A \overset{*}{\Longrightarrow} w_{i,j}\}$, where $w_{i,j}$ is the substring from positions i to j. For example, for string *baaba*, $w_{1,1} = b$ and $w_{2,4} = aab$.

{S, A, C}	← $X_{1,5}$			
Ø	{S, A, C}			
Ø	{B}	{B}		
{S, A}	{B}	{S, C}	{S, A}	
{B}	{A, C}	{A, C}	{B}	{A, C}
b	**a**	**a**	**b**	**a**