Theory of Computation

Fall 2018, Midterm Exam.

Nov. 13, 2018

---

1. (15 pts) True or False? Justify your answer in a brief yet convincing way. No penalty for wrong answers.

   (1) Let $half(L) = \{x \mid \exists y \in \Sigma^*, \text{ with } |y| \in \{|x|, |x|+1\} \text{ and } xy \in L\}$. If $half(L)$ is context-free, then L must be context-free.
   Sol. False. $\{a^n b^{2n} c^n \mid n \geq 0\}$

   (2) Suppose L_1 is a regular language, then $L_2 = \{ww^R \mid w \in L_1\}$ is always a context-free language.
   Sol. True. Let M be an FA accepting L_1. Design a PDA M' that simulates (nondeterministically) M on the first half w of the input while pushing w onto the stack. Then use the stack to compare with the rest (i.e., w^R) of the input.

   (3) The language $a^*b^*c^* - \{a^n b^n c^n \mid n \geq 0\}$ is context-free.
   Sol. True. Check (nondeterministically) $(\#a \neq \#b)$ or $(\#a \neq \#c)$ or $(\#b \neq \#c)$ using a stack.

   (4) Let L be a language and h a homomorphism. If $h(L)$ is regular, then L must be regular.
   Sol. False. $h(a) = \epsilon$. $h(\{a^{n^2}\}) = \epsilon$.

   (5) Suppose $\Sigma = \{0, 1\}$, and let $sort(x)$ be the function that reorders the symbols in x in numerical order. Let $sort(L) = \{sort(x) \mid x \in L\}$. For example, if $L = \{0, 1, 01, 10, 101, 0110\}$, then $sort(L) = \{0, 1, 01, 011, 0011\}$. Regular languages are closed under sort.
   L is regular, it is the set of all strings of length >=3 with same start and end letter
   so, no R_L doesnt have infinite index
   Sol. False. Consider $L = \{(01)^n \mid n \geq 0\}$. Sort$(L)$=$\{0^n 1^n \mid n \geq 0\}$.

2. (10 pts) Recall that the R_L relation (over Σ^*) associated with language L is defined to be xR_Ly iff $\forall z \in \Sigma^*, xz \in L \Leftrightarrow yz \in L$. For language $L = \{x \in \{0,1\}^* \mid \ x = wcw^R,\ |c| > 0,\ |w| > 0\}$, does R_L have infinitely many or finitely many distinct equivalence classes? In the former case, you need not describe all equivalence classes of R_L, just show that there are infinitely many of them. In the latter case, enumerate all the classes. (Note: An equivalence class C is represented as $[w]$, where w is the shortest word in C.)
   Sol. Since c can be an arbitrary string of length > 0, every $w \in L$ is of the form $0c0$ or $1c1$ for some $|c| > 0$. R_L includes $\{\epsilon\}$; $\{0\}$; $\{1\}$; $\{00, 01, 0c1 \mid c \in \Sigma^+\}$; $\{10, 11, 1c0 \mid c \in \Sigma^+\}$; $\{0c0 \mid c \in \Sigma^+\}$; $\{1c1 \mid c \in \Sigma^+\}$

3. (10 pts) Prove that the following language is not regular by applying the pumping lemma. x=a^(k+1) and y=a^k z=a
   $L = \{a^i b^j \mid i, j \geq 1, (i \geq j) \text{ or } (i < j \text{ and } j \text{ is a multiple of } i)\}$
   Sol. Let $z = a^{3k} b^{3k}$. Suppose$a^{3k}b^{3k} = uvw$, such that $|uv| \leq k$, $|v| > 0$. Since $|uv| \leq k$, v contains only a's. Let $v = a^m, 1 \leq m \leq k$. $uv^0w = a^{3k-m}b^{3k}$ is not in the language as $3k$ is not a multiple of $3k - m$.

4. (10 pts) Design an algorithm to decide, given a CFG $G = (V, \Sigma, P, S)$ in Chomsky Normal Form, whether exists an $x \in L(G)$ such that $|x|$ is even.
   Hint: create a set Δ (which is empty initially) and assign A_{ev} and A_{od} for each $A \in V$ to Δ iteratively such that for some w,

   - $A_{ev} \in \Delta$ if $A \stackrel{*}{\Rightarrow} w$ and $|w|$ is even; and
   - $A_{od} \in \Delta$ if $A \stackrel{*}{\Rightarrow} w$ and $|w|$ is odd.

   the set of all strings of even length is regular, take intersection with the PDA for G, we can then check if the resultant grammar is empty by checking the nullability of the start symbol -> can be dome when the grammar is in chomsky normal form

   Sol. Initially, $\Delta = \emptyset$

   For each $A \Rightarrow a$, add A_{od} to Δ

   For each $A \Rightarrow \epsilon$, add A_{ev} to Δ

   Repeat the following until Δ does not grow

   For each $A \to BC$

   - $B_{ev}, C_{ev} \in \Delta$, add A_{ev} to Δ
   - $B_{ev}, C_{oh} \in \Delta$, add A_{od} to Δ
   - $B_{od}, C_{ev} \in \Delta$, add A_{od} to Δ
   - $B_{od}, C_{od} \in \Delta$, add A_{ev} to Δ

take intersection with a*b*, this results in {a^nb^n}

5. (5 pts) Consider the following language over the alphabet $\Sigma = \{a, b, c\}$. $L = \{a^n b^p (c+b)^{n-p} \mid 1 \leq n, 1 \leq p \leq n\}$, where $(c+b)^{n-p}$ means a sequence of $n-p$ symbols from the set $\{c, b\}$. Prove that L is not regular using the closure properties of regular languages. Do not use the pumping lemma.
Sol. $h(a) = a; h(b) = b; h(c) = b$. $h(L) = \{a^n b^n \mid n \geq 1\}$

6. (6 pts) The quotient L_1/L_2 of two languages L_1 and L_2 is defined as $L_1/L_2 = \{x \mid \exists y \in L_2, xy \in L_1\}$. Let $L_1 = \{w \in \{0,1\}^* \mid w$ has an even number of $0's\}$, $L_2 = \{0\}$, $L_3 = \{0, 00\}$. Answer the following questions:

   (a) What is L_1/L_2? L1/L2 = {x | x has an odd no. of 0s}

   (b) What is L_1/L_3? L1/L3 = sigma*

   Sol. $L_1/L_2 = \{w \in \{0,1\}^* | w$ has an odd number of 0's$\}$, $L_1/L_3 = \{0,1\}^*$.

7. (10 pts) The following $\{0, 1, 2, 3\}$-valued function F for regular expressions is defined recursively as follows:

   - $F(\emptyset) = 0$, $F(\epsilon) = 1$;
   - $F(a) = 2$, for each $a \in \Sigma$,
   - $F(\sigma \cup \tau) = max\{F(\sigma), F(\tau)\}$;
   - If $F(\sigma) = 0$ or $F(\tau) = 0$ then $F(\sigma \cdot \tau) = 0$ else $F(\sigma \cdot \tau) = max\{F(\sigma), F(\tau)\}$;
   - If $F(\sigma) \leq 1$ then $F(\sigma^*) = 1$ else $F(\sigma^*) = 3$.

   Answer the following questions, and justify your answers:

   (a) What is $F(((00 \cup 11)^* \cdot \emptyset) \cup (11 \cup 22))$? Why?
   Sol. $F(((00 \cup 11)^* \cdot \emptyset) \cup (11 \cup 22)) = max\{F((00 \cup 11)^* \cdot \emptyset), F(11 \cup 22)\} = max\{0, 2\} = 2$.

   (b) What kind of regular expressions would make $F(\sigma) = 3$? Why?
   Sol. $L(\sigma)$ is an infinite set.

8. (9 pts) Is the language $\{x \in \{a, b\}^* \mid |x|$ is even and the first half of x has one more "a" than does the second half$\}$ context-free? Justify your answer formally.
Sol. Not context-free. If L were context-free, then $L' = L \cap a^*b^*a^*b^*$ would also be context-free. But it is not, which we show using the Pumping Theorem. Consider $w = a^{k+1}b^k a^k b^{k+1}$, where k is the pumping constant. Suppose $w = a^{k+1}b^k a^k b^{k+1} = uvxyz$. Divide w into four regions (a's, then b's, then a's, then b's). Consider the following cases:

   - If either v or y crosses a region boundary, pump in. The resulting string is not in L because the characters will be out of order.
   - If $|vy|$ is not even, pump in once. The resulting string will not be in L because it will have odd length.
   - Now we consider all the cases in which neither v nor y crosses regions and $|vy|$ is even. In what follows, case (a, b) denote that v in region a and y in region b.

   (1, 1) pump in once. The boundary between the first half and the second half shifts to the left. Hence, the first half has more as.

   (2, 2) pump out. Since $|vy|$ is even, we pump out at least 2 b's so at least one a migrates from the second half to the first.

   (3, 3) pump out. This decreases the number of a's in the second half. Only bs migrate in from the first half.

   (4, 4) pump in once. The boundary between the first half and the second half shifts to the right, causing a's to flow from the second half into the first half.

   ... the remaining cases can be argued similarly.

9. (10 pts) Let $Half(L) = \{x \mid \exists y \in \Sigma^*, |x| = |y|, xy \in L\}$. Consider $L = \{0^i 1^j 2^j 3^{3i} \mid i, j \geq 1\} (\subseteq \{0, 1, 2, 3\}^*)$. Answer the following questions:

   yes, context free, can make grammar for it

   (a) (3 pts) Is L context-free? Why?
   Sol. Yes.

   (b) (3 pts) What is $Half(L)$?
   Sol. $\{0^i 1^j 2^i \mid j \geq i\} \cup \{0^i 1^j 2^j 3^{i-j} \mid j < i\}$

(c) (4 pts) Is $Half(L)$ a CFL? Why?
Sol. $Half(L) \cap 0^*1^*2^* = \{0^n1^m2^n \mid m, n \geq 0, m \geq n\}$ – not context-free

10. (15 pts) Consider the binary operator $\diamond$ on languages defined as follows: given two languages L_1 and L_2 over Σ, $L_1 \diamond L_2$ consists of words of the form uv such that $u \in L_1$, $v \in L_2$, and $|u| = |v|$.

(a) (10 pts) Prove that if L_1 and L_2 are regular languages, then $L_1 \diamond L_2$ is a CFL. To this end, Let $M_i = (Q_i, \Sigma, \delta_i, q_i, F_i)$ be an FA accepting L_i, $i \in \{1, 2\}$. Construct a PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$ to accept $L_1 \diamond L_2$. Show your construction in detail.
Sol. the set of states of M is $Q_1 \cup Q_2 \cup \{f\}$. The set of stack symbols is $\{Z, Z_0\}$. The initial state of M is q_1. For $q \in Q_1$ and a symbol a and a stack symbol X, M contains a transition $(\delta_1(q, a), ZX)) \in \delta(q, a, X)$. For $q \in F_1$ and a stack symbol X, M contains a transition $(q_2, X) \in \delta(q, \epsilon, X)$. For $q \in Q_2$ and a symbol a, M contains a transition $(\delta_2(q, a), \epsilon) \in \delta(q, a, X)$. For $q \in F_2$, M contains a transition $(f, \epsilon) \in \delta(q, \epsilon, Z_0)$.

(b) (5 pts) Give examples to show that if L_1 is a regular language and L_2 is a CFL then $L_1 \diamond L_2$ need NOT be a CFL.
Sol. $L_1 = a^*$; $L_2 = \{b^nc^n \mid n \geq 0\}$. Then $L_1 \diamond L_2 = \{a^{2n}b^nc^n \mid n \geq 0\}$– not context-free.