CS21004 Formal Languages and Automata Theory, Spring 2012–13

Mid-Semester Test

Maximum marks: 34 | Date: 22-Feb-2013 | Duration: Two hours

Roll no: ____________________ **Name:** ____

[*Write your answers in the question paper itself. Be brief and precise. Answer all questions.*]

1. Let $L_1 = \mathcal{L}(S)$ and $L_2 = \mathcal{L}(T)$, where the non-terminal symbols S and T satisfy the productions:

$$\begin{array}{rcl} S & \rightarrow & a \mid b \mid abSS. \\ T & \rightarrow & a \mid b \mid TTab. \end{array}$$

Find examples of strings α of length forty such that:

(a) $\alpha \in L_1$ and $\alpha \in L_2$. $(ab)^{20}$

(b) $\alpha \in L_1$ and $\alpha \notin L_2$. $(ab)^{19}ba$

(c) $\alpha \notin L_1$ and $\alpha \in L_2$. $ba(ab)^{19}$

(d) $\alpha \notin L_1$ and $\alpha \notin L_2$. a^{40}

For each part, only one example of length forty suffices. **(4)**

2. Convert the following grammar (over the alphabet $\{a, b, c, d\}$) to the Chomsky normal form. **(6)**

$$\begin{array}{rcl} S & \rightarrow & aSd \mid T, \\ T & \rightarrow & bTc \mid \epsilon. \end{array}$$

Solution Throwing out the ϵ-production $T \rightarrow \epsilon$ requires throwing in the two productions $T \rightarrow bc$ and $S \rightarrow \epsilon$. Throwing out the new ϵ-production $S \rightarrow \epsilon$ requires throwing in the production $S \rightarrow ad$. Finally, throwing out the unit production $S \rightarrow T$ requires throwing in the productions $S \rightarrow bTc$ and $S \rightarrow bc$. The rest is simple. The grammar in the Chomsky normal form is given below.

$$\begin{array}{rcl} S & \rightarrow & AD \mid BC \mid AU \mid BV, \\ T & \rightarrow & BC \mid BV, \\ U & \rightarrow & SD, \\ V & \rightarrow & TC, \\ A & \rightarrow & a, \\ B & \rightarrow & b, \\ C & \rightarrow & c, \\ D & \rightarrow & d. \end{array}$$

3. Two strings α, β of the *same length* over the alphabet $\{0, 1\}$ are said to have *Hamming distance* k if they differ in exactly k positions. For example, the strings 1101110010 and 1101011010 have Hamming distance two, because they differ only in the fifth and the seventh positions. Let L be a language. By $H_k(L)$, we define the language consisting of strings α such that α is at Hamming distance k from some string in L. If L is regular, prove that $H_k(L)$ is regular for each fixed $k \geqslant 0$. **(6)**

(**Warning:** You are your friend's friend. So be careful.)

Solution Let D be a DFA accepting L. We construct an NFA N to accept $H_k(L)$. We start with $k + 1$ copies of D, call them $D^{(0)}, D^{(1)}, \ldots, D^{(k)}$. The only start state of N is the start state of the copy $D^{(0)}$, and the final states of N are only the final states in the copy $D^{(k)}$. Whenever there is a transition from p to q in D, marked by 0, make a transition, marked by 1, from $p^{(i)}$ to $q^{(i+1)}$ for all $i = 0, 1, 2, \ldots, k-1$. Likewise, whenever there is a transition from p to q in D, marked by 1, make a transition, marked by 0, from $p^{(i)}$ to $q^{(i+1)}$ for all $i = 0, 1, 2, \ldots, k-1$. The only way N can accept is by entering a final state in $D^{(k)}$ after reading the entire input. But that requires exactly k bit flips in the input for an accepting string for D.

Notice that proceeding by induction on k may lead to problems. As basis cases, you may prove the statement for $k = 0, 1$, and then tend to argue that $H_{k+1}(L) = H_1(H_k(L))$. But this is incorrect. We actually have $H_1(H_k(L)) = H_{k+1}(L) \cup H_{k-1}(L)$. However, this union need not be disjoint, and so you cannot say $H_{k+1}(L) = H_1(H_k(L)) \setminus H_{k-1}(L)$.

Roll no: ________________ **Name:** ________________________________

4. A context-free grammar is called strongly *right linear* if each production in the grammar is of the form $A \to aB$ or $A \to \epsilon$, where A, B are non-terminal symbols and a is a terminal symbol. Prove that L is the language of a strongly right-linear grammar if and only if L is regular. **(6)**

Solution Let $L = \mathcal{L}(D)$ for some DFA $D = (Q, \Sigma, \delta, s, F)$. Design a strongly right linear grammar (N, Σ, P, S) accepting L as follows. Take $N = Q$ and $S = s$. For every transition $\delta(p, a) = q$, add the production $p \to aq$. Finally, for each final state $f \in F$, add the production $f \to \epsilon$. Every sentential form (which is not a sentence) in a derivation contains exactly one non-terminal symbol at the end, preceded by the symbols read from the input. Finally, a non-terminal symbol representing a final state vanishes to give a sentence which is a string accepted by D.

Conversely, let (N, Σ, P, S) be a strongly right-linear grammar having language L. We construct an NFA $N = (Q, \Sigma, \Delta, T, F)$ to accept L as follows. We take $Q = N$, $T = \{S\}$, and F to be those non-terminal symbols A for which $A \to \epsilon$ is a production. Finally, for every production of the form $A \to aB$, we make a transition of N from A to B, marked by a.

5. One of the following two languages is context-free, and the other is not. Identify which one is what. Justify. **(6+6)**

(a) $L_a = \{a^l b^m c^n \mid l, m, n \geqslant 0,\ l + m \geqslant n\}$.

(b) $L_b = \{a^l b^m c^n \mid l, m, n \geqslant 0,\ l \geqslant n \text{ and } m \geqslant n\}$.

Solution L_a is context-free. A simple way to prove this is by designing a context-free grammar to accept L_a. Here it is with the start symbol S.

$$\begin{array}{rcl} S & \to & aSc \mid aS \mid T, \\ T & \to & bTc \mid bT \mid \epsilon. \end{array}$$

L_b is not context-free. We prove this using the pumping lemma. Suppose that L_b is context-free, and let k be a pumping-lemma constant for L_b. Feed the string $\beta = a^k b^k c^k \in L_b$ to the pumping lemma. We obtain a decomposition of the form $\beta = \beta_1\beta_2\beta_3\beta_4\beta_5$ such that at least one of β_2 and β_4 is non-empty, $|\beta_2\beta_3\beta_4| \leqslant k$, and $\gamma^{(i)} = \beta_1\beta_2^i\beta_3\beta_4^i\beta_5 \in L_b$ for all $i \geqslant 0$. Now, consider all possible cases.

Case 1: *Either β_2 or β_4 runs across the a, b or the b, c boundary.*

In this case, $\gamma^{(2)}$ is not of the form $a^*b^*c^*$. So, we will henceforth assume that each of β_2 and β_4 belongs to a single block. Since they cannot be widely apart, we consider only the two following cases.

Case 2: *Both β_2 and β_4 belong to the same block.*

If they belong to the block of a's or the block of b's, then $\gamma^{(0)}$ contains more c's than allowed. If they belong to the block of c's, then $\gamma^{(2)}$ contains more c's than allowed.

Case 3: *β_2 and β_4 belong to two adjacent blocks.*

If these blocks are of a's and b's, consider $\gamma^{(0)}$ which has more c's than either a's or b's (or both). If β_2 belongs to the block of b's and β_4 to the block of c's, we consider two subcases. If β_4 is empty, we look at $\gamma^{(0)}$ which contains less b's than c's. Finally, if $\beta_4 \neq \epsilon$, then $\gamma^{(2)}$ contains less a's than c's.

In all the cases, we can produce a string (either $\gamma^{(0)}$ or $\gamma^{(2)}$) which is both outside L (by definition) and inside L (by the pumping lemma). This contradiction proves that L_b is not context-free.

[Space for leftover answers and rough work]

[Space for leftover answers and rough work]