Midterm II

CS 373: Theory of Computation

Date: Tuesday, November 4, 2008.

Instructions:

- This is a closed book exam. No notes, cheat sheets, textbook, or printed material allowed.
- You have 120 minutes to solve this exam.
- There are 6 problems in this exam, each worth 10 points. However, not all problems are of equal difficulty.
- Please write your name on the top of *every* page in the space provided.
- If your solution does not fit in the space provided, and continues onto one of the back sheets, please indicate clearly where we should look for the solution.
- Unless otherwise stated, recall that "prove that", "show that" for a problem means you need to formally prove what you are claiming.
- Answering "I don't know" for a problem *does not receive any points.*

Name	SOLUTIONS
Netid	

Problem	Maximum Points	Points Earned	Grader
1	10		
2	10		
3	10		
4	10		
5	10		
6	10		
Total	60		

Problem 1. **True/False.** Decide for each statement whether it is true or false. Circle **T** if the statement is *necessarily true*; circle **F** if it it is not necessarily true. In addition, provide a one line justification for your answer. Each correct answer with the right justification earns **2 points**, while each correct answer without the right justification earns only **1 point**.

(a) $L = \{a^i b^j c^k d^k e^j f^i \mid i, j, k \geq 0\}$ is not context-free.
T **F**

False. There are many ways of seeing this. Consider the grammar $S \rightarrow aSf \mid A$, $A \rightarrow bAe \mid B$, $B \rightarrow cBd \mid \epsilon$; this grammar accepts L.

(b) $L = \{a^n b^m a^n \mid m = n \bmod 5\}$ is context free. Recall $a = b \bmod c$ iff a and b leave the same remainder when divided by c.
T **F**

True. Here is a CFG for this language. $S \rightarrow a^5 S c^5 \mid abc \mid aabbcc \mid aaabbbccc \mid aaaabbbbcccc \mid \epsilon$.

(c) Stack space of a PDA P on input w is the maximum number of symbols on the stack at any time during any computation of P on w. For any P, there is a PDA P' such that $L(P) = L(P')$ and on any input $w \in L(P)$, the stack space of P' on w is half the stack space of P on w.
T **F**

True. As mentioned in the newsgroup, the stack symbols of P' will be a pair of stack symbols of P; so each time P' pushes or pops, it will push or pop two symbols of P. P' will simulate P by also storing on stack symbol in the control state.

(d) Let L be a context-free language and let R be a regular language. Then $L \setminus R$ is context-free.
T **F**

True. $L \setminus R = L \cap (\bar{R})$ and so since regular languages are closed under complementation, and context-free languages are closed under intersection with regular languages, the result follows.

(e) Let L be a context-free language and let R be a regular language. Then $R \setminus L$ is context-free.
T **F**

False. If it were true, then context-free languages would be closed under complementation (take $R = \Sigma^*$), which we know to not hold.

Problem 2. Consider

$$L = \{w\#x \mid \exists u,v.\ x = uw^Rv,\ \text{and } u,v,w,x \in \{0,1\}^*\}$$

Prove that L is context-free by either constructing a CFG G with $L = L(G)$, or a PDA P with $L = L(P)$. You need not formally prove that your construction is correct. However, you should clearly outline the intuition behind your construction, by either spelling out the strings generated from the variables of your grammar, or giving an informal pseudo-code outlining how your PDA works.

Solution: This problem is the same as problem 2.6(c) (and 2.7(c)) which has solutions in the book. □

Problem 3.

Recall that $L^R = \{w \mid w^R \in L\}$ where w^R is the reverse of the string w. We can show that context free languages are closed under this operation as follows: Given a "normalized" PDA P recognizing L, construct a PDA P' (also normalized) that, for any input w, accepts w iff P accepts w^R, by emulating the operations of P in reverse. Complete the following proof of this closure property based on the above plan.

Let $P = (Q, \Sigma, \Gamma, \delta, q_0, F)$ be a "normalized" PDA accepting the language L, that satisfies the following conditions.

- P has exactly one accept state, i.e., $F = \{q_F\}$ for some $q_F \in Q$
- P empties its stack before accepting, i.e., if $\langle q_0, \epsilon \rangle \xrightarrow{w}_P \langle q_F, \sigma \rangle$ then $\sigma = \epsilon$.

The PDA recognizing L^R is given by $P' = (Q', \Sigma, \Gamma, \delta', q_0', F')$ where

(a) $Q' = \underline{Q}$ **[1 point]**

(b) $q_0' = \underline{q_F,\ \text{where } F = \{q_F\}}$ **[1 point]**

(c) $F' = \underline{\{q_0\}}$ **[1 point]**

(d) δ' is defined in terms of δ as follows: $(q_2, \sigma) \in \delta'(q_1, \rho, a)$ iff $\underline{(q_1, \rho) \in \delta(q_2, \sigma, a)}$

Thus, P' pushes what P pops, and pops what P pushes **[2 points]**

(e) Another way to prove that CFLs are closed under reversing is by constructing a grammar to recognize L^R. Formally, let $G = (V, \Sigma, R, S)$ be a CFG such that $L(G) = L$. Define $G^R = (V, \Sigma, R', S)$ such that $R' = \{A \to \alpha^R \mid A \to \alpha \in R\}$. Thus, G^R has the same variables, terminals and start symbol as G, and the rules in G^R are those of G with the right-side reversed. Prove that for every $A \in V$ and $w \in \Sigma^*$, $A \overset{*}{\Rightarrow}_G w$ iff $A \overset{*}{\Rightarrow}_{G^R} w^R$. **[4 points]**

We will prove this by induction on the number of steps in the derivation $A \overset{*}{\Rightarrow}_G w$. For the base case, consider a one step derivation $A \overset{*}{\Rightarrow}_G w$. Now, $A \overset{*}{\Rightarrow}_G w$ in one step iff $A \to w \in R$ iff $A \to w^R \in R'$ (by definition of R') iff $A \overset{*}{\Rightarrow}_{G^R} w^R$ in one step.

For the induction hypothesis, assume that for every $A \in V$ and $w \in \Sigma^*$, $A \overset{*}{\Rightarrow}_G w$ in k steps (where $k < n$) iff $A \overset{*}{\Rightarrow}_{G^R} w^R$ in k steps.

Consider A, w such that $A \overset{*}{\Rightarrow}_G w$ in n steps. Now such a derivation must be of the form $A \Rightarrow_G X_1 X_2 \cdots X_k \overset{*}{\Rightarrow}_G w$, where $X_i \in \Sigma \cup V$. Now given this derivation, it must be the case that there exist $w_1, w_2, \ldots w_k$ such that $w_i = X_i$ if $X_i \in \Sigma$, $w_i \in \Sigma^*$ if $X_i \in V$, $w = w_1 w_2 \cdots w_k$, and $X_i \overset{*}{\Rightarrow}_G w_i$ (if $X_i \in V$) in less than n steps. By induction hypothesis, we have $X_i \overset{*}{\Rightarrow}_{G^R} w_i$ (if $X_i \in V$). Therefore, we have the following derivation in G^R.

$$A \Rightarrow_{G^R} X_k X_{k-1} \cdots X_2 \overset{*}{\Rightarrow}_{G^R} w_k^R X_{k-1} \cdots X_1 \overset{*}{\Rightarrow}_{G^R} \cdots \overset{*}{\Rightarrow}_{G^R} w_k^R w_{k-1}^R \cdots w_1^R = (w_1 \cdots w_k)^R = w^R$$

(f) Use the result in part (e), to show that $L(G^R) = L^R$. **[1 point]**

From part (e), we know $S \overset{*}{\Rightarrow}_G w$ iff $S \overset{*}{\Rightarrow}_{G^R} w^R$. Thus $w \in L(G)$ iff $w^R \in L(G^R)$.

Problem 4. Consider

$$L = \{(a^n b^n)^n \mid n \geq 0\}$$

Prove that L is not context-free.

Solution: Using Closure Properties: Consider homomorphism $h_1 : \{a, b, c, d, e\} \to \{a, b\}^*$ where

$$h_1(a) = h_1(c) = h_1(e) = a \qquad h_1(b) = h_1(d) = b$$

Let $L_1 = h_1^{-1}(L) = \{((a \cup b \cup e)^n (b \cup e)^n)^n \,|\, n \geq 0\}$. Take $L_2 = L_1 \cap a^* b^* c^* (d^* e^*)^* = \{a^n b^n c^n d^n (e^n d^n)^{n-2} \,|\, n \geq 0\}$. Now consider a hommorphism $h_2 : \{a, b, c, d, e\} \to \{a, b, c\}^*$ where

$$h_2(a) = a \qquad h_2(b) = b \qquad h_2(c) = c \qquad h_2(d) = h_2(e) = \epsilon$$

Now $L_3 = h_2(L_2) = \{a^n b^n c^n \mid n \geq 0\}$ is not context-free. Thus, L is not context-free since L_3 is obtained from L by apply context-free-preserving operations.

Pumping Lemma Proof: Let p be the pumping length. Pick $z = (a^p b^p)^p \in L$. Let u, v, w, x, y be some division of z such that $z = uvwxy$, $|vwx| \leq p$, and $|vx| > 0$. Now $uv^2wx^2y \notin L$; this can be proved in two ways.

- **Combinatorial Argument:** Notice that any string in L is of the form $(a^n b^n)^n$ and so has length $2n^2$. Now $|uv^2wx^2y| > 2p^2$, because $|vx| > 0$. Further $|uv^2wx^2y| \leq 2p^2 + p$ since $|vwx| \leq p$. Thus, $2p^2 < 2p^2 + 1 \leq |uv^2wx^2y| \leq 2p^2 + p \leq 2p^2 + 2p + 2 = 2(p+1)^2$ and so $uv^2wx^2y \notin L$.
- **Standard Case-by-Case Analysis:** We consider 4 cases based on what form vwx takes.
  - $vwx \in L(a^*)$, i.e., consists only of as. Then, $u = (a^p b^p)^s a^i$, $v = a^{j_1}$, $w = a^{j_2}$, $x = a^{j_3}$, and $y = a^k b^p (a^p b^p)^t$, where $s + t = p - 1$, $i + j_1 + j_2 + j_3 + k = p$ and $j_1 + j_2 > 0$. Now, $uv^2wx^2y = (a^p b^p)^s a^i a^{2j_1} a^{j_2} a^{2j_3} a^k b^p (a^p b^p)^t = (a^p b^p)^s (a^{p+|vx|} b^p)(a^p b^p)^t \notin L$.
  - $vwx \in L(a^* b^*)$, i.e., begins with as and ends with bs. Then $u = (a^p b^p)^s a^i$, $vwx = a^{j_1} b^{j_2}$, and $y = b^k (a^p b^p)^t$, where $s + t = p - 1$, and $j_1 + j_2 \neq p$ and $|vx| > 0$. Now in uv^2wx^2y, if we consider the block of $a^* b^*$ of which vwx is a substring, will contain either more than p as or bs (or both). Thus, $uv^2wx^2y \notin L$.
  - $vwx \in L(b^*)$. This case is similar to case 1.
  - $vwx \in L(b^* a^*)$. This case is similar to case 2.

□

Problem 5. Prove that if G is a CFG in Chomsky Normal Form, then for any string $w \in L(G)$ of length $n \geq 1$, exactly $2n - 1$ steps are required for any derivation of w.

Solution: Proof 1: The result from the following sequence of observations. First if $w \in L(G)$ then there is a parse with yield w. Next, any parse tree of a grammar in Chomsky Normal Form is a binary tree. Since the yield is w, the tree has n leaves, which means that it has $2n - 1$ vertices. Finally, the number of derivation steps is equal to the number of vertices in the parse tree. Thus, the result follows.

Proof 2: One can also prove this by induction on the length of w. More precisely, we will show that for any $A \in V$ and $w \in \Sigma^*$, if $A \stackrel{*}{\Rightarrow} w$ then A derives w in $2|w| - 1$ steps.

Base Case: Consider $|w| = 1$. Since G is in Chomsky normal form, it does not have any nullable variables, or unit productions, if $A \stackrel{*}{\Rightarrow} w$ then $A \rightarrow w$ is a rule. Thus, the derivation is of length $1 = 2(1) - 1$.

Induction Hypothesis: If $A \stackrel{*}{\Rightarrow} w$ for any w such that $|w| < k$ then A derives w in $2|w| - 1$ steps.

Induction Step: Consider $A \stackrel{*}{\Rightarrow} w$ for $|w| = k$. Since $k > 1$, and G is in Chomsky Normal Form, it must be the case that $A \Rightarrow BC \stackrel{*}{\Rightarrow} w$. Now, if $BC \stackrel{*}{\Rightarrow} w$ and G has no nullable symbols, there must be u, v such that $w = uv$, $B \stackrel{*}{\Rightarrow} u$, $C \stackrel{*}{\Rightarrow} v$, and u and v is not ϵ. Thus, $|u| < k$ and $|v| < k$. By induction hypothesis, B derives u in $2|u| - 1$, and C derives v in $2|v| - 1$. Thus, A derives w in $1 + 2|u| - 1 + 2|v| - 1 = 2(|u| + |v|) - 1 = 2k - 1$.
□

Problem 6. For a Language $L \subseteq \Sigma^*$ define reflect$(L) = \{ww^R \mid w \in L\}$ where w^R denotes the reverse of the string w.

(a) Consider $L_0 = \{0^i 1^i \mid i \geq 0\}$ and $L_1 = L(0^*1^*)$, languages over $\{0,1\}$. What is reflect(L_0) and reflect(L_1)? **[4 points]**

reflect$(L_0) = \{0^i 1^{2i} 0^i \mid i \geq 0\}$ and reflect$(L_1) = \{0^i 1^{2j} 0^i \mid i,j \geq 0\}$. *Note:* reflect$(L_1) \neq L(0^*1^*1^*0^*)$, because in reflect$(L_1)$ the number of 1s in the middle is even, and the number of 0s at the beginning and end are equal.

(b) Prove that context-free languages are not closed under the operation reflect$(\cdot)$. In other words, there is a context-free language L such that reflect(L) is not context-free. **[6 points]**

Clearly, L_0 is context-free (proved in class). We will show that reflect(L_0) is not context-free. One can prove this in a number of ways. The pumping lemma proof is the same as that in the solutions of problem 1 in homework 7. Here we present a proof using closure properties to give a different way of looking at it. Consider the homomorphism $h_1 : \{a,b,c\} \to \{0,1\}^*$ defined as follows:

$$h_1(a) = 0 \qquad h_1(b) = 11 \qquad h_1(c) = 0$$

Now $L_1 = h_1^{-1}(L) = \{(a \cup c)^n b^n (a \cup c)^n \mid n \geq 0\}$. Take $L_2 = L_1 \cap a^*b^*c^* = \{a^n b^n c^n \mid n \geq 0\}$, which we know is not context-free.