

Introduction to the Theory of Computation

Jean Gallier

Homework 5

March 25, 2004; Due April 8, 2004

“A problems” are for practice only, and should not be turned in.

Problem A1. Give constructions proving that for any PDA M , the acceptance modes $T(M)$, $N(M)$, and $L(M)$, are equivalent.

Problem A2. (i) Given any set X , for any two subsets $A, B \subseteq X$, prove that $A \subseteq B$ iff $A \cap \overline{B} = \emptyset$, where \overline{B} denotes the complement of B in X .

(ii) Sketch an algorithm to test whether $L(G) \subseteq L(D)$, where G is a context-free grammar and D is a DFA.

Problem A3. Given a homomorphism $h: \Sigma^* \rightarrow \Delta^*$, for any any context-free language $L \subseteq \Delta^*$, prove that $h^{-1}(L)$ is context-free.

Hint. Find a construction using a PDA.

“B problems” must be turned in.

Problem B1 (40 pts). Use the pumping lemma (or Ogden’s lemma) to show that the following languages are not context-free:

$$L_1 = \{a^m b^n c^p \mid 1 \leq m < n < p\}$$

$$L_2 = \{a^n b^n c^p \mid n, p \geq 1, p \neq n\}$$

Hint. For L_1 , pick w with the a ’s be distinguished. For L_2 , pick w with the c ’s distinguished.

Problem B2 (50 pts). Give pushdown automata for the following languages:

(a) $L_5 = \{ww^R \mid w \in \{a, b\}^*\}$ (w^R denotes the reversal of w)

(b) $L_6 = \{a^m b^n \mid 1 \leq m \leq n \leq 3m\}$

(c) $L_7 = \{a^n b^n \mid n \geq 1\} \cup \{a^{2n} b^n \mid n \geq 1\}$

(d) $L_8 = \{a^{2m} b^n a^{2m} b^p \mid m, n, p \geq 1\} \cup \{a^m b^{3n} a^p b^{3n} \mid m, n, p \geq 1\}$

(e) $L_9 = \{xycy \mid |x| = |y|, x, y \in \{a, b\}^*\}$

In each case, give a brief justification of the fact that your PDA generates the desired language.

Problem B3 (30 pts). (1) Given the alphabet $\Sigma_2 = \{a, b, \bar{a}, \bar{b}\}$, define the relation \simeq on Σ_2^* as follows: For all $u, v \in \Sigma_2^*$,

$$u \simeq v \quad \text{iff} \quad \exists x, y \in \Sigma_2^*, \quad u = xa\bar{a}y, \quad v = xy \quad \text{or} \quad u = x\bar{b}by, \quad v = xy.$$

Let \simeq^* be the reflexive and transitive closure of \simeq , and let $D_2 = \{w \in \Sigma_2^* \mid w \simeq^* \epsilon\}$. Give a context-free grammar for D_2 , and justify your answer.

Note: Strings such as $a\bar{a}b\bar{b}$ and $ab\bar{b}\bar{a}$ are in D_2 .

(2) Given the alphabet $\Sigma_m = \{a_1, \dots, a_m, \bar{a}_1, \dots, \bar{a}_m\}$, define the relation \simeq on Σ_m^* as follows: For all $u, v \in \Sigma_m^*$,

$$u \simeq v \quad \text{iff} \quad \exists x, y \in \Sigma_m^*, \quad u = xa_i\bar{a}_iy, \quad v = xy, \quad \text{for some } i, 1 \leq i \leq m.$$

Let \simeq^* be the reflexive and transitive closure of \simeq , and let $D_m = \{w \in \Sigma_m^* \mid w \simeq^* \epsilon\}$. Give a context-free grammar for D_m , and justify your answer.

Note: D_m is known as the *Dyck set* on m letters.

Problem B4 (60 pts). A context-free grammar, $G = (V, \Sigma, P, S)$, is *linear* iff for every production $(A \rightarrow \alpha) \in P$,

$$\alpha \in \Sigma^* N \Sigma^* \cup \Sigma^*,$$

where $N = V - \Sigma$. A language, L , is a *linear context-free language* iff there is some linear context-free grammar, G , such that $L = L(G)$.

(a) Recall that for any string, $w \in \Sigma^*$, the string w^R denotes the reversal of the string w , i.e., the string w written in reverse order ($\epsilon^R = \epsilon$). Prove that the language $L_0 = \{ww^R \mid w \in \{a, b\}^*\}$ is linear context-free.

(b) Given any regular language, $L \subseteq \Sigma^*$, prove that $L^R = \{w^R \mid w \in L\}$ is also regular.

(c) Let $G = (V, \Sigma, P, S)$ be a *linear context-free* grammar. Assume that the set of productions P contains p productions and that it is ordered in some fashion. Each production is named p_i , with $1 \leq i \leq p$. Let $\Delta = \{\delta_1, \dots, \delta_p\}$ be a set in one-to-one correspondence with the set of productions P and assume that Δ is disjoint from V . The language $R \subseteq \Delta^*$ is defined as follows:

$$R = \{\delta_{i_1} \cdots \delta_{i_n} \mid n \geq 1, \quad \text{there is a derivation } S \implies \alpha_1 \implies \cdots \implies \alpha_n, \\ \text{such that } \alpha_n \in \Sigma^* \quad \text{and the production applied at the } k\text{-th step is } p_{i_k}\}.$$

Let $h_1, h_2: \Delta^* \rightarrow \Sigma^*$ be the homomorphisms defined as follows: For any $p_i \in P$,

- (1) If p_i is of the form $A \rightarrow uBv$, where $A, B \in N$, and $u, v \in \Sigma^*$, then $h_1(\delta_i) = u$ and $h_2(\delta_i) = v$;
- (2) If p_i is of the form $A \rightarrow w$, where $A \in N$ and $w \in \Sigma^*$, then choose any $u, v \in \Sigma^*$ such that $w = uv$, and let $h_1(\delta_i) = u$ and $h_2(\delta_i) = v$.

Prove that for any k , with $1 \leq k \leq n - 1$,

$$\alpha_k = h_1(\delta_{i_1} \cdots \delta_{i_k}) B h_2(\delta_{i_k} \cdots \delta_{i_1}),$$

for some $B \in N$, and that

$$\alpha_n = h_1(\delta_{i_1} \cdots \delta_{i_n}) h_2(\delta_{i_n} \cdots \delta_{i_1}).$$

Conclude that $L(G) = \{h_1(w)h_2(w^R) \mid w \in R\}$, for some language R over Δ .

(d) Prove that the language R defined in (c) is regular.

(e) Given a linear context-free language, $L = L(G)$, from questions (c) and (d), we know that there is a regular language R and two homomorphisms h_1, h_2 , such that

$$L = \{h_1(w)h_2(w^R) \mid w \in R\}.$$

Let $\Omega = \{\omega_1, \dots, \omega_p\}$ be a set in one-to-one correspondence with Δ and such that Ω is disjoint from V and Δ . Let $f: \Delta^* \rightarrow \Omega^*$ be the homomorphism determined by defining $f(\delta_i) = \omega_i$, for all i , with $1 \leq i \leq p$. Let S be the regular language

$$S = \{f(w)^R \mid w \in R\} = f(R)^R,$$

where R is the regular set of question (c). Also, let $g_1: (\Delta \cup \Omega)^* \rightarrow \Sigma^*$ be the homomorphism determined by defining

$$g_1(\delta_i) = h_1(\delta_i) \quad \text{and} \quad g_1(\omega_i) = h_2(\delta_i).$$

Finally, let $g_2: (\Delta \cup \Omega)^* \rightarrow \{a, b\}^*$ be the homomorphism determined by defining

$$g_2(\delta_i) = a^i b \quad \text{and} \quad g_2(\omega_i) = b a^i,$$

for all i , with $1 \leq i \leq p$.

Prove that

$$g_2^{-1}(L_0) \cap RS = \{w_1 w_2 \mid w_1 \in R, \quad w_2 \in S, \quad \text{and} \quad f(w_1) = w_2^R\},$$

where L_0 is the language of question (a), R is defined in question (c), and S is defined above in (e). From this, prove that

$$L = g_1(g_2^{-1}(L_0) \cap RS).$$

Hence, prove that every linear context-free language can be obtained from the language $L_0 = \{w w^R \mid w \in \{a, b\}^*\}$ by homomorphism, inverse homomorphism, and intersection with regular languages.

Remark: In fact, it can be shown that the family of linear context-free languages is the least class of languages with these properties.

Extra Credit (50 pts). Using (e), prove that the language $\{a^m b^m a^n b^n \mid m, n \geq 1\}$ is not linear-context-free.

Hint. First, prove that if $h_i: \Sigma \rightarrow \Delta_i$ are homomorphisms ($i = 1, 2$), c is a new symbol not in $\Sigma \cup \Delta_1 \cup \Delta_2$, and $R \subseteq \Sigma^*$ is a regular language, then $\{h_1(w)ch_2(w^R) \mid w \in R\}$ is a linear context-free language. Then, use some suitable gsm mappings. This is hard!

Problem B5 (80 pts). In this problem, the fundamental property of LR-parsing (due to Knuth) is established.

For simplicity, let us consider context-free grammars without ϵ -rules. Given a reduced context-free grammar $G = (V, \Sigma, P, S')$ augmented with start production $S' \rightarrow S$, where S' does not appear in any other productions, the set C_G of *characteristic strings of G* is the following subset of V^* (watch out, not Σ^*):

$$C_G = \{\alpha\beta \in V^* \mid S' \xRightarrow{rm}^* \alpha B v \xRightarrow{rm} \alpha\beta v, \\ \alpha, \beta \in V^*, v \in \Sigma^*, B \rightarrow \beta \in P\}.$$

In words, C_G is a certain set of prefixes of sentential forms obtained in rightmost derivations: those obtained by truncating the part of the sentential form immediately following the rightmost symbol in the righthand side of the production applied at the last step.

The fundamental property of LR-parsing is that C_G is a *regular language*. A nondeterministic automaton N_{C_G} accepting C_G can be constructed according to the method described in Section 1 of the handout *A Survey of LR-Parsing Methods, etc.*. Please, review this construction.

(i) Let G be the following grammar:

$$\begin{aligned} S' &\rightarrow E \\ E &\rightarrow E + T \mid T \\ T &\rightarrow T * F \mid F \\ F &\rightarrow (E) \mid a \end{aligned}$$

with $\Sigma = \{+, *, (,), a\}$.

Give the automaton N_{C_G} for the grammar G .

(ii) Using the standard algorithms, give a deterministic finite automaton equivalent to N_{C_G} . Do not include the “dead state”.

(iii) You shall now prove that $L(N_{C_G}) = C_G$! This proves the correctness of the method that you are going to implement in the programming project!

(1) Prove the following claim by induction on the length of rightmost derivations:

Claim 1: For any nonterminal A , for every rightmost derivation

$$A \xRightarrow{rm}^* \alpha B v \xRightarrow{rm} \alpha\beta v,$$

where $v \in \Sigma^*$, $B \in N$, and $\alpha, \beta \in V^*$, if we denote the first production in the above rightmost derivation as $A \rightarrow \delta$, then there is a computation on input $\alpha\beta$ from state $A \rightarrow \cdot$ to the final state $B \rightarrow \beta \cdot$.

To prove this claim, you will have to show the following (think about it in terms of parse trees). For any nonterminal A , every rightmost derivation from A is either of the form

- (i) $A \xRightarrow{rm} \delta$, for some production $A \rightarrow \delta$, in which case $A = B$ and $\delta = \beta$, or of the form
- (ii) $A \xRightarrow{rm} \lambda B_i \rho \xRightarrow{rm}^* \lambda B_i w \xRightarrow{rm}^* \lambda \alpha_i B w_i w \xRightarrow{rm} \lambda \alpha_i \beta w_i w$, with $w, w_i \in \Sigma^*$, $A, B, B_i \in N$, $\lambda, \rho, \alpha_i, \beta \in V^*$, and where

$$B_i \xRightarrow{rm}^* \alpha_i B w_i \xRightarrow{rm} \alpha_i \beta w_i \quad \text{and} \quad \rho \xRightarrow{rm}^* w.$$

Let $B_i \rightarrow \delta_i$ be the first production applied in the rightmost derivation from B_i . In the first case, there is a computation in N_{C_G} from state $A \rightarrow \cdot$ to the final state $A \rightarrow \delta \cdot$ (where again, $A \rightarrow \delta = B \rightarrow \beta$), and in the second case, there is a computation in N_{C_G} from state $A \rightarrow \cdot$ to $B_i \rightarrow \cdot$ on input λ , and a computation from state $B_i \rightarrow \cdot$ to the final state $B \rightarrow \beta \cdot$ on input $\alpha_i \beta$.

Conclude that C_G is a subset of $L(N_{C_G})$.

(2) Prove the following claim by induction on the number of ϵ -transitions in a computation in N_{C_G} :

Claim 2: For any state $A \rightarrow \cdot$, if there is a computation on input γ to some final state $B \rightarrow \beta \cdot$, then there is some rightmost derivation $A \xRightarrow{rm}^* \alpha B v \xRightarrow{rm} \alpha \beta v$, such that, the production applied in the first rightmost derivation step is $A \rightarrow \delta$, and $\gamma = \alpha \beta$.

For this, prove the following:

- (i) Either $\gamma = \delta$ and the computation is from state $A \rightarrow \cdot$ to state $A \rightarrow \delta \cdot$, or
- (ii) δ is of the form $\lambda B_i \rho$, γ is of the form $\lambda \alpha_i \beta$, and there is a computation on input $\alpha_i \beta$ from some state of the form $B_i \rightarrow \cdot$ to the final state $B \rightarrow \beta \cdot$, and a rightmost derivation as in Claim 1.

Conclude that $L(N_{C_G})$ is a subset of C_G , thus establishing that $C_G = L(N_{C_G})$.

TOTAL: 260 points + 50 points.