# Automata Theory - Homework II (Solutions)

K. Subramani
LCSEE,
West Virginia University,
Morgantown, WV
{ksmani@csee.wvu.edu}

## 1 Problems

1. Let $L$ be a regular language not containing $\lambda$. Argue that there exists a right-linear grammar for $L$, whose productions are restricted to the forms:

$$\begin{array}{rcl} A & \rightarrow & aB, \text{ and} \\ A & \rightarrow & a \end{array}$$

   where $A$ and $B$ are generic variables and $a$ is a generic terminal.

   **Solution:** In class, we showed that every regular language can be represented as a DFA $M = \langle Q, \Sigma, \delta, q_0, F \rangle$, where the symbols should be interpreted using the standard convention. Since, $L$ is $\lambda$-free, it must be the case that $q_0 \notin F$. We construct the following Right-Linear Grammar $G = \langle V, T, S, P \rangle$ for $L$:

   (a) We use symbol $A_i$ to denote state $q_i$. $V = \{A_i : i = 0, 1, \ldots, |Q| - 1\}$.

   (b) The set of terminals $T$ is precisely $\Sigma$.

   (c) $S = A_0$.

   (d) The productions $P$ of $G$ are defined as follows: For each transition, $\delta(q_i, a) = q_j$, add the production $A_i \rightarrow aA_j$. If $q_j$ is also a final state, then add the production $A_i \rightarrow a$.

   We use induction on the number of transition steps to show that if $\delta(q_0, w) \in F$ , then $A_0 \Rightarrow^* w$. Likewise, we use induction on the number of steps in a leftmost derivation to establish that if $A_0 \Rightarrow^* w$, then $\delta(q_0, w) \in F$.
   (The induction proofs are straightforward exercises). $\Box$

2. Consider the language $L = \{a^n : \ n \text{ is not a perfect square}\}$. Prove that $L$ is not regular, *by using the Pumping Lemma.* You may not use complement properties of regular languages.

   **Solution:** One way of showing that $L$ is not regular is through the following argument: If $L$ is regular, then so is $L^c$; however, we showed in class that $L^c$ is not regular and hence $L$ cannot be regular. Since you were expressly forbidden from using the complement properties of regular languages, let us proceed to prove the non-regularity of $L$, using first principles.

   Assume that $L$ is regular and let $M = \langle Q, \Sigma, \delta, q_0, F \rangle$ denote the unique minimal state DFA for $L$, where the symbols in the tuple have their usual meaning. Let $p$ denote the number associated by the Pumping Lemma with $L$; we know that $p = |Q|$. Let $G = \{a^{p^2+i} : i = 1, 2, \ldots 2p\}$; we can think of the strings in $G$ as being indexed by their length after subtracting the offset $p^2$. Using $w_i$ to refer to the string $a^{p^2+i}$, we have that $|w_i| + 1 = |w_{i+1}|$.
   Let $R = \{\delta(q_0, w_i) : w_i \in G\}$; we use $r_i$ to denote $\delta(q_0, w_i), \ w_i \in G$.

   The following observations are in order:

   (i) All the states in $R$ are final states.

(ii) $|G| = 2p$ and $|R| \leq p$. As per the pigeonhole principle, there must exist at least two strings $w_i$ and $w_j$ in $G$, such that $\delta(q_0, w_i) = \delta(q_0, w_j) = r_k$, $r_k \in R$.

We have thus established that there exists a string $a^s, 1 \leq s \leq p$, and a state $r_k \in R$, such that $\delta(r_k, a^s) = r_k$. Let $T = \{\delta(r_k, a), \delta(r_k, a^2), \ldots, \delta(r_k, a^{s-1}), \delta(r_k, a^s)\}$; clearly $T \subseteq R$. $T$ represents a cycle of final states on $a$-transitions; it follows that the DFA can never escape from this cycle towards a non-final state on $a$-transitions. Thus $\delta(q_0, a^{p^2+2p+1}) \in T$; however, $a^{p^2+2p+1} = a^{(p+1)^2}$ and should be rejected by the DFA $M$.

We thus have the desired contradiction and it follows that $L$ cannot be a regular language.

$\Box$

3. Consider the grammar $G = \langle V, T, S, P \rangle$, with productions defined by:

$$S \rightarrow aSbS \mid bSaS \mid \lambda$$

Is $G$ ambiguous? Is $L(G)$ ambiguous?

**Solution:** $G$ is ambiguous, since the string $w = abab$ has two distinct leftmost derivations:

(i) $S \Rightarrow aSbS \Rightarrow abSaSbS \Rightarrow abaSbS \Rightarrow ababS \Rightarrow abab$, and

(ii) $S \Rightarrow aSbS \Rightarrow abS \Rightarrow abaSbS \Rightarrow ababS \Rightarrow abab$.

$L(G)$ is the language of strings over $\{a, b\}$, in which the number of $a$s is equal to the number of $b$s. An unambiguous grammar for this language is given by: $G' = \langle V, T, S, P \rangle$, where,

(a) $V = \{S\}$,

(b) $T = \{a, b\}$,

(c) $S = S$, and

(d) $P$ is defined by:

$$S \rightarrow aSb \mid bSa \mid S \cdot S \mid \lambda$$

You are required to use induction on the number of steps of a leftmost derivation from $S$ to establish that if $S \Rightarrow^*_{lm} w$, then the leftmost derivation of $w$ is unique. $\Box$

4. Show that the language $L = \{w \cdot w^R \ : \ w \in \{a, b\}^*\}$ is not inherently ambiguous.
*Hint: Prove that $L$ has an unambiguous grammar.*

**Solution:** An unambiguous grammar for $L$ is $G = \langle V, T, S, P \rangle$, where,

(a) $V = \{S\}$.

(b) $T = \{a, b\}$.

(c) $S = S$.

(d) The productions $P$ are defined by:

$$S \quad \rightarrow \quad aSa \mid bSb \mid \lambda$$

In order to establish the unambiguous nature of $G$, we need to show that for every string $w \in L(G)$, there is precisely one leftmost derivation $S \Rightarrow^*_{lm} w$; this is done by using induction on the length of $w$. Before commencing the proof, we need the following lemmata.

***Lemma 1.1*** *If $S \Rightarrow^*_{lm} w$, then $|w|$ is even.*

**Proof:** Exercise. Use induction on the number of steps in the *shortest* leftmost derivation of $w$. $\Box$

***Lemma 1.2*** *If $w \in L$, then the string $w'$ obtained by dropping the first and last symbols of $w$, also belongs to $L$*

**Proof:** It is a straightforward observation (Use contradiction!). $\Box$

***Theorem 1.1*** *If $S \Rightarrow^*_{lm} w$, then this derivation is unique.*

**Proof:** We use induction on the length of $w$; the induction will be on the set of even numbers and not on the set of natural numbers (which is our conventional ground set). If $|w| = 0$, then $w$ must be $\lambda$, and there is precisely one production rule and hence one way for $w$ to be derived from $S$, in leftmost fashion.

Assume that Theorem (1.1) is true, for all even-length strings of length at most $2 \cdot k,\ k \geq 0$. Now consider a string $w \in L$, of length $2 \cdot (k + 1)$. Since $w \in L$, it must be the case that $w = axa$ or $w = bxb$, with $x \in L$ (See Lemma (1.2)). However, $|x| = 2 \cdot k$ and hence $x$ has a unique leftmost derivation from $S$. It therefore follows that $w$ has a unique leftmost derivation; for instance, if $w = axa$, then $S \Rightarrow_{lm} aSa \Rightarrow^*_{lm} axa$ represents the unique leftmost derivation of $w$ from $S$. A similar argument holds when $w = bxb$; we can therefore apply the principle of mathematical induction to conclude that every string in $L$ has a unique leftmost derivation from $S$, i.e., $G$ is unambiguous. Inasmuch as $L$ has an unambiguous grammar, it follows that $L$ itself is unambiguous. $\Box$

$\Box$

5. Remove all unit productions, $\lambda$-productions and useless productions from the the grammar $G = \langle V, T, P, S \rangle$, with productions $P$ defined by:

$$\begin{aligned} S &\rightarrow aA \mid aBB \\ A &\rightarrow aaA \mid \lambda \\ B &\rightarrow bB \mid bbC \\ C &\rightarrow B \end{aligned}$$

**Solution:** It is important that all $\lambda$-productions are deleted first, followed by the unit productions and finally by the useless productions. If this order is altered, language preservation is not guaranteed [Lin06].

(a) Removing $\lambda$-productions - The only nullable symbol is $A$; accordingly, applying the algorithm in [Lin06], the removal of $\lambda$-productions results in the following set of productions:

$$\begin{aligned} S &\rightarrow aA \mid aBB \mid a \\ A &\rightarrow aaA \mid aa \\ B &\rightarrow bB \mid bbC \\ C &\rightarrow B \end{aligned}$$

(b) Removing unit productions - There is precisely one unit production, viz., $C \rightarrow B$. Applying the algorithm in [Lin06], the removal of this unit production results in the following set of productions:

$$\begin{aligned} S &\rightarrow aA \mid aBB \mid a \\ A &\rightarrow aaA \mid aa \\ B &\rightarrow bB \mid bbB \end{aligned}$$

(c) Removing useless productions - Observe that no terminal string can be derived from $B$; it follows that any production involving $B$ can be deleted, without affecting the language of the grammar. Accordingly, the final set of productions is:

$$\begin{aligned} S &\rightarrow aA \mid a \\ A &\rightarrow aaA \mid aa \end{aligned}$$

$\Box$

## References

[Lin06] Peter Linz. *An Introductionto Formal Languages and Automata*. Jones and Bartlett, $4^{th}$ edition, 2006.