Homework 8

John Carlyle

November 22, 2013

1.

b.

$$S \to SS \mid bS \mid a$$

$\{x \mid x$ does not end in $b\}$

c.

$$S \to SaS \mid b$$

$\{x \mid x$ contains alternating a and b starting and ending with $b\}$

e.

$$S \to TT$$
$$T \to Ta \mid aT \mid b$$

$\{x \mid n_b(x) = 2\}$

f.

$$S \to aSa \mid bSb \mid aAb \mid bAa$$
$$A \to aAa \mid bAb \mid a \mid b \mid \lambda$$

$\{x \mid x \neq aa, bb, \lambda\}$

g.

$$S \to aT \mid bT \mid \lambda$$
$$T \to aS \mid bS$$

$\{x \mid |x| = \text{even }\}$

h.

$$S \to aT \mid bT$$
$$T \to aS \mid bS \mid \lambda$$

$\{x|\ |x| = \text{odd}\ \}$

3.

a. The set of odd-length strings in $\{a, b\}^*$ with middle symbol a.

$$S \to aSa \mid aSb \mid bSa \mid bSb \mid a$$

b. The set of even-length strings in $\{a, b\}^*$ with two middle symbols equal.

$$S \to aSa \mid aSb \mid bSa \mid bSb \mid aa \mid bb$$

c. The set of odd-length strings in $\{a, b\}^*$ whose first, middle and last symbols are the same.

$$S \to aAaAa \mid bAbAb \mid a \mid b$$
$$A \to aA \mid bA \mid \lambda$$

4.

a.

Let G be the grammar defined by $S \to SabS \mid SbaS \mid \lambda$.
Prove that $\forall x \in L(G)$ that $n_a(x) = n_b(x)$

Proof. (by structural induction)

Base Case:

The only derivation with one step is $S \Rightarrow \lambda$
$\therefore w = \lambda$ and $n_a(\lambda) = n_b(\lambda)$, So the base case holds.

Inductive Hypothesis:

Assume for all derivations $S \Rightarrow^* w$ where $|w| = n \geq 1$ steps that $n_a(w) = n_b(w)$.

Inductive Step:

We must show for every derivation $S \Rightarrow^* w$ with $n+1$ steps that $n_a(w) = n_b(w)$.
Let $S \Rightarrow^* w$ be a derivation with $n + 1$ steps. Since $n + 1 > 1$, the first step cannot be $S \Rightarrow \lambda$
$\therefore$ the first step must be $S \Rightarrow SabS$ or $S \Rightarrow SbaS$.

Case 1:

$w = w_1abw_2$ where w_1, w_2 are derived from S in the remaining n steps.

$\therefore$ by the I.H. $w_1, w_2 \in L(G)$

$\therefore n_a(w) = n_a(w_1ab_w2) = n_a(w_1) + n_a(w_2) + 1$

$\therefore n_b(w) = n_b(w_1ab_w2) = n_b(w_1) + n_b(w_2) + 1$

But because of I.H. $n_a(w_1) + n_a(w_2) = n_b(w_1) + n_b(w_2)$

So we are left with two values that are the same, each with 1 added to it.

$\therefore\ n_a(w) = n_b(w)$

Case2:

Case 1 can be repeated without loss of generality. The order of a and b are simply switched and addition is commutative.

□

Example of a string in $L(G)$ but not in the grammar: *aabb*

b.

Let G be the grammar defined by $S \to aSb \mid bSa \mid abS \mid baS \mid Sab \mid Sba \mid \lambda$. Prove that $\forall x \in L(G)$ that $n_a(x) = n_b(x)$

Proof. (by structural induction)

Base Case:

The only derivation with one step is $S \Rightarrow \lambda$

$\therefore w = \lambda$ and $n_a(\lambda) = n_b(\lambda)$, So $w \in L(G)$.

Inductive Hypothesis:

Assume for all derivations $S \Rightarrow^* w$ where $|w| = n \geq 1$ steps that $w \in L(G)$.

Inductive Step:

We must show for every derivation $S \Rightarrow^* w$ with $n+1$ steps that $w \in L(G)$.

Let $S \Rightarrow^* w$ be a derivation with $n+1$ steps. Since $n+1 > 1$, the first step cannot be $S \Rightarrow \lambda$

$\therefore$ the first step must be $S \Rightarrow aSb$ or $S \Rightarrow bSa$ or $S \Rightarrow abS$ or $S \Rightarrow baS$ or $S \Rightarrow Sab$ or (finally) $S \Rightarrow Sba$.

Case 1:

$w = aw_1b$ where w_1 is derived from S in the remaining n steps.

$\therefore$ by the I.H. $w_1 \in L(G)$

$\therefore n_a(w) = n_a(aw_1b) = n_a(w_1) + 1$

$\therefore n_b(w) = n_b(aw_1b) = n_b(w_1) + 1$

But because of I.H. $n_a(w_1) = n_b(w_1)$

So we are left with two values that are the same, each with 1 added to it.

$\therefore\ w \in L(G)$

Case 2-6:

These can be repeated without loss of generality. Which case only affects which order the terms are added together. Because addition is commutative this changes nothing meaningful.

$\square$

Example of a string in $L(G)$ but not in the grammar: Since the G has all possible permutations of a, b, S G is capable of generating any possible set of strings with the property $n_b = n_a$.

7. Describe the language generated by the CFG with productions:

$$S \rightarrow ST \mid \lambda$$
$$T \rightarrow aS \mid bT \mid b$$

The above CFG describes the language $L = \{a, b\}^*$

Proof. $L(G) \subseteq \{a, b\}^*$
This is trivially true since any language that uses $\Sigma = \{a, b\}$ is going to be a subset of $\{a, b\}^*$ $\square$

Proof. $L(G) \supseteq \{a, b\}^*$

Base Case: Let $w = \lambda$ then w can be derived by using the production $S \Rightarrow \lambda$ and $\lambda \in \{a, b\}^*$ hence the base case holds.

Inductive Hypothesis: Assume $w \in \{a, b\}^*$ and $w \in L$.

Inductive Step: We must show that $x\sigma \in L(G)$ where $\sigma \in \{a, b\}$. Since G has only two terminating symbols (λ, b), and σ can only be a or b, hence there are four cases.

Case 1: The rightmost derivation of w is $S \rightarrow \lambda$ and $\sigma = a$
Since $S \Rightarrow \lambda$ is the rightmost derivation: $\alpha S \Rightarrow \alpha\lambda$. We can replace this derivation with $\alpha S \Rightarrow \alpha ST \Rightarrow \alpha\lambda T \Rightarrow \alpha\lambda aS \Rightarrow \alpha\lambda a\lambda \Rightarrow \alpha a$ hence $w\sigma \in L(G)$.

Case 2: The rightmost derivation of w is $S \rightarrow \lambda$ and $\sigma = b$
Since $S \Rightarrow \lambda$ is the rightmost derivation: $\alpha S \Rightarrow \alpha\lambda$. We can replace this derivation with $\alpha S \Rightarrow \alpha ST \Rightarrow \alpha Sb \Rightarrow \alpha\lambda b \Rightarrow \alpha b$ hence $w\sigma \in L(G)$.

Case 3: The rightmost derivation of w is $S \rightarrow b$ and $\sigma = a$
Since $\alpha T \Rightarrow \alpha b$. We can replace this derivation with $\alpha T \Rightarrow \alpha bT \Rightarrow \alpha baS \Rightarrow \alpha ba\lambda \Rightarrow \alpha ba$. hence $w\sigma \in L(G)$.

Case 4: The rightmost derivation of w is $S \rightarrow b$ and $\sigma = b$
Sinc $\alpha T \Rightarrow \alpha b$. We can replace this derivation with $\alpha T \Rightarrow \alpha BT \Rightarrow \alpha bT \Rightarrow \alpha ba$ hence $w\sigma \in L(G)$.

Hence every $w\sigma \in L(G)$.

$\square$

10.

a.

$$
\begin{aligned}
S &\Rightarrow AB \mid \lambda \\
A &\Rightarrow aAB \mid a \mid \lambda \\
B &\Rightarrow bB \mid b
\end{aligned}
$$

b.

$$
\begin{aligned}
S &\Rightarrow ABB \mid \lambda \\
A &\Rightarrow aAB \mid a \mid \lambda \\
B &\Rightarrow bB \mid b
\end{aligned}
$$

21. Proof that every language that is regular, is also a CFL.

Proof. Let R be the set of regular languages.

Base Case: There are two base cases:

1. $\lambda \in R$ by definition. λ can be represented by a CFG with the single production $S \to \lambda$ Hence this case holds.
2. Let $\sigma \in \Sigma$. $\sigma \in R$ by definition. Similar to case 1: σ can be represented by a CFG with the single production rule $S \to \sigma$. Hence the second base case holds.

Inductive Hypothesis: Suppose L_1 and L_2 are regular languages that can also be represented by a CFG.

Inductive Step: There are three cases:

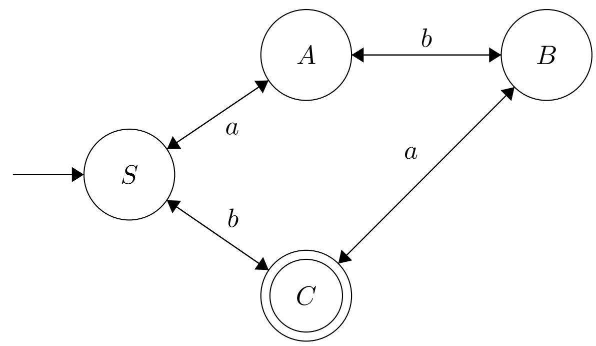
1. $L_1 L_2 \in R$ by definition of regular languages. By the Inductive Hypothesis L_1, L_2 are both expressable by CFG as well. And by Theorem 4.9 $L_1 L_2$ is also expressable by some CFG.
2. $L_1 \cup L_2 \in R$ by definition of regular languages. By the Inductive Hypothesis L_1, L_2 are both expressable by CFG as well. And by Theorem 4.9 $L_1 \cup L_2$ is also expressable by some CFG.
3. $L_1^* \in R$ by definition of regular languages. By the Inductive Hypothesis L_1 is expressable by a CFG as well. And by Theorem 4.9 L_1^* is also expressable by some CFG.

Hence all ways of buiding regular languages also build CFGs. $\square$

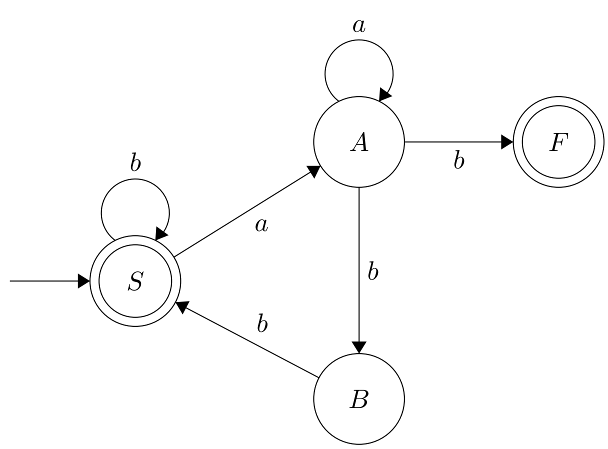
23. Not done because the material was not taught. I have about 3 pages of notes and many hours of wasted time.

26.

a.



b.



27. Figure 4.33 in the book represented as a grammar.

$$A \Rightarrow aB \mid bD \mid \lambda$$
$$B \Rightarrow aB \mid bC$$
$$C \Rightarrow aB \mid bC \mid \lambda$$
$$D \Rightarrow aD \mid bD$$