CS150 Homework 4 Solution

Please read the text book [1] and the references [2].

Q1

(a)

\[
\begin{align*}
S & \rightarrow AB \\
A & \rightarrow 0A1 \mid e \\
B & \rightarrow 1B0 \mid e
\end{align*}
\]

(b)

\[
\begin{align*}
S & \rightarrow AB \mid CD \\
A & \rightarrow 00A1 \mid e \\
B & \rightarrow 1B00 \mid e \\
C & \rightarrow 00C1 \mid 01 \\
D & \rightarrow 1D00 \mid 0
\end{align*}
\]

Q2

\[
\begin{align*}
S & \rightarrow A1B \rightarrow 0A1B \rightarrow 01B \rightarrow 010B \rightarrow 0101B \rightarrow 01010B \\
& \rightarrow 010101B \rightarrow 010101
\end{align*}
\]

Please see Figure 1. It is the parse tree.

Ex. 5.4.5

(a)

If there is only one leftmost derivation for any given string, then the grammar is unambiguous. So we only need to show that there is only one leftmost derivation for a string in the language.

When we start with leftmost derivations, we can only start with the first rule, because that is the only one we get. When we encounter symbol A, we can look ahead at the current symbol in the input string (we scan the input string from left to right as in the example of the lecture notes). If the current symbol is 0, then we can only apply the rule \( A \rightarrow 0A \). If the symbol is 1, then we have to apply the rule \( A \rightarrow \epsilon \). So, there is no ambiguity here. When we encounter symbol B, similar reasoning can again be applied, but we observe the rule \( B \rightarrow \epsilon \) can only be applied at the end of the input. So there is only one leftmost derivation for any given input and the language is unambiguous.
(b)
We just change the B rules and the grammar can become ambiguous.

\[ B \rightarrow BB \mid 0 \mid 1 \mid \text{epsilon} \]

**Ex. 6.1.1**

(b)

For string 0011:

ID 1: \( (q, 0011, Z_0) \rightarrow (q, 011, XZ_0) \rightarrow (q, 11, XXZ_0) \rightarrow (q, 1, XXZ_0) \rightarrow (p, \text{epsilon}, XXZ_0) \rightarrow (p, \text{epsilon}, XZ_0) \rightarrow (p, \text{epsilon}, Z_0) \)

ID 2: \( (q, 0011, Z_0) \rightarrow (q, 011, XZ_0) \rightarrow (q, 11, XXZ_0) \rightarrow (q, 1, XXZ_0) \rightarrow (p, 1, XZ_0) \rightarrow (p, 1, Z_0) \rightarrow (p, \text{epsilon}, \epsilon) \)

ID 3: \( (q, 0011, Z_0) \rightarrow (q, 011, XZ_0) \rightarrow (q, 11, XXZ_0) \rightarrow (q, 1, XXZ_0) \rightarrow (p, 1, XZ_0) \rightarrow (p, \text{epsilon}, XXZ_0) \rightarrow (p, \text{epsilon}, XZ_0) \rightarrow (p, \text{epsilon}, Z_0) \)

ID 4: \( (q, 0011, Z_0) \rightarrow (q, 011, XZ_0) \rightarrow (p, 011, Z_0) \)

ID 5: \( (q, 0011, Z_0) \rightarrow (q, 011, XZ_0) \rightarrow (q, 11, XXZ_0) \rightarrow (p, 11, XZ_0) \rightarrow (p, 1, XXZ_0) \rightarrow (p, \text{epsilon}, XXXZ_0) \)
For string 010:

ID 1: (q, 010, Z_0) |- (q, 10, XZ_0) |- (q, 0, XZ_0) |- (q, epsilon, XXZ_0) |- (p, epsilon, XZ_0) |- (p, epsilon, Z_0)

ID 2: (q, 010, Z_0) |- (q, 10, XZ_0) |- (q, 0, XZ_0) |- (p, 0, Z_0)

ID 3: (q, 010, Z_0) |- (q, 10, XZ_0) |- (p, 10, Z_0) |- (p, 0, epsilon)

Q5

(a)

We use accept by final state: X is used to track the number of 1s, and Y is used to track the number of 0s, q is the start state and p is the accept state.

\[
\begin{align*}
\delta(q, 1, Z_0) &= \{ (q, XZ_0) \} \\
\delta(q, 0, Z_0) &= \{ (q, YZ_0) \} \\
\delta(q, 1, X) &= \{ (q, XX) \} \\
\delta(q, 1, Y) &= \{ (q, \text{epsilon}) \} \\
\delta(q, 0, X) &= \{ (q, \text{epsilon}) \} \\
\delta(q, 0, Y) &= \{ (q, YY) \} \\
\delta(q, \text{epsilon}, X) &= \{ (p, X) \} \\
\delta(q, \text{epsilon}, Y) &= \{ (p, \text{epsilon}) \} \\
\delta(p, \text{epsilon}, X) &= \{ (r, X) \} \\
\delta(p, \text{epsilon}, Y) &= \{ (r, Y) \} \\
\delta(r, \text{epsilon}, Z_0) &= \{(r, \text{epsilon})\}
\end{align*}
\]

(b)

We use accept by empty stack.

\[
\begin{align*}
\delta(q, 0, Z_0) &= \{ (q, XZ_0) \} \\
\delta(q, 0, X) &= \{ (q, XX) \} \\
\delta(q, 1, Z_0) &= \{ (p, Y) \} \\
\delta(q, 1, X) &= \{ (p, e) \} \\
\delta(p, 0, Y) &= \{ (r, e) \} \\
\delta(r, e, Z_0) &= \{ (r, e) \} \\
\delta(q, 1, Y) &= \{ (q, YY) \} \\
\delta(p, 1, X) &= \{ (p, e) \} \\
\delta(r, 0, Y) &= \{ (r, e) \}
\end{align*}
\]

Here, e means epsilon.
References
