Albert-Ludwigs-Universität, Inst. für Informatik
Prof. Dr. Fabian Kuhn
Mohamad Ahmadi

Theoretical Computer Science - Bridging Course
Winter Term 2019/2020
Exercise Sheet 4

for getting feedback submit electronically by 12:15, Monday, November 18, 2019

Exercise 1: Context-Free Grammar *(3+2 Points)*

For each of the following languages, give a context-free grammar to accept the language.

(a) $L_1 = \{w\#w' | w^R \text{ is a substring of } w', \text{ and } w, w' \in \{0,1\}^*\}$. [1]

(b) $L_2 = \{0^i 1^j 2^k | i \neq j \text{ or } j \neq k\}$

Sample Solution

(a) $S \to AB$
$A \to 0A0 \mid 1A1 \mid \#B$
$B \to 0B \mid 1B \mid \varepsilon$

Each word in language L_1 is in the form of $w\#xw^R y$, where $x, y \in \{a,b\}^*$. Then, variable B generates x and y, and variable A generates $w\#xw^R$.

(b) $S \to AC \mid BC \mid DE \mid DF$
$A \to 0 \mid 0A \mid 0A1$
$B \to 1 \mid B1 \mid 0B1$
$C \to 2 \mid 2C$
$D \to 0 \mid 0D$
$E \to 1 \mid 1E \mid 1E2$
$F \to 2 \mid F2 \mid 1F2$

Exercise 2: Chomsky Normal Form *(5 Points)*

Consider the following context-free grammar (CFG):

$$S \to aSb \mid D$$
$$D \to ccDcc \mid \varepsilon$$

Convert this CFG into an equivalent one in Chomsky Normal Form. Give the grammar you obtained after each step of the conversion algorithm.

[1] w^R is achieved by reversing the order of the symbols in w.

Sample Solution

Add a new start variable S_0 and the rule $S_0 \to S$.

$$\begin{aligned} S_0 &\to S \\ S &\to aSb \mid D \\ D &\to ccDcc \mid \varepsilon \end{aligned}$$

Remove all ε-rules: Delete the rule $D \to \varepsilon$ and add the rules $S \to \varepsilon$ and $D \to cccc$.

$$\begin{aligned} S_0 &\to S \\ S &\to aSb \mid D \mid \varepsilon \\ D &\to ccDcc \mid cccc \end{aligned}$$

Remove $S \to \varepsilon$ and add $S \to ab$ and $S_0 \to \varepsilon$ (the ε-rule for the start variable is allowed).

$$\begin{aligned} S_0 &\to S \mid \varepsilon \\ S &\to aSb \mid ab \mid D \\ D &\to ccDcc \mid cccc \end{aligned}$$

Next remove unit rules.
Remove $S_0 \to S$ and add $S_0 \to aSb \mid ab \mid D$.

$$\begin{aligned} S_0 &\to \varepsilon \mid aSb \mid ab \mid D \\ S &\to aSb \mid ab \mid D \\ D &\to ccDcc \mid cccc \end{aligned}$$

Remove $S_0 \to D$ and add $S_0 \to ccDcc \mid cccc$.

$$\begin{aligned} S_0 &\to \varepsilon \mid aSb \mid ab \mid ccDcc \mid cccc \\ S &\to aSb \mid ab \mid D \\ D &\to ccDcc \mid cccc \end{aligned}$$

Remove $S \to D$ and add $S \to ccDcc \mid cccc$.

$$\begin{aligned} S_0 &\to \varepsilon \mid aSb \mid ab \mid ccDcc \mid cccc \\ S &\to aSb \mid ab \mid ccDcc \mid cccc \\ D &\to ccDcc \mid cccc \end{aligned}$$

Convert the rules into the proper form.
Add $S_1 \to Sb$ and adjust the rules accordingly.

$$\begin{aligned} S_0 &\to \varepsilon \mid aS_1 \mid ab \mid ccDcc \mid cccc \\ S &\to aS_1 \mid ab \mid ccDcc \mid cccc \\ S_1 &\to Sb \\ D &\to ccDcc \mid cccc \end{aligned}$$

Add $U_1 \to a$ and Add $U_2 \to b$ and adjust.

$$
\begin{aligned}
S_0 &\to \varepsilon \mid U_1S_1 \mid U_1U_2 \mid ccDcc \mid cccc \\
S &\to U_1S_1 \mid U_1U_2 \mid ccDcc \mid cccc \\
S_1 &\to SU_2 \\
U_1 &\to a \\
U_2 &\to b \\
D &\to ccDcc \mid cccc
\end{aligned}
$$

Add $S_2 \to cS_3$, $S_3 \to DS_4$ and $S_4 \to cc$ and adjust.

$$
\begin{aligned}
S_0 &\to \varepsilon \mid U_1S_1 \mid U_1U_2 \mid cS_2 \mid cccc \\
S &\to U_1S_1 \mid U_1U_2 \mid cS_2 \mid cccc \\
S_1 &\to SU_2 \\
S_2 &\to cS_3 \\
S_3 &\to DS_4 \\
S_4 &\to cc \\
U_1 &\to a \\
U_2 &\to b \\
D &\to cS_2 \mid cccc
\end{aligned}
$$

Add $S_5 \to cS_6$ and $S_6 \to cc$ and adjust.

$$
\begin{aligned}
S_0 &\to \varepsilon \mid U_1S_1 \mid U_1U_2 \mid cS_2 \mid cS_5 \\
S &\to U_1S_1 \mid U_1U_2 \mid U_3S_2 \mid cS_5 \\
S_1 &\to SU_2 \\
S_2 &\to cS_3 \\
S_3 &\to DS_4 \\
S_4 &\to cc \\
S_5 &\to cS_6 \\
S_6 &\to cc \\
U_1 &\to a \\
U_2 &\to b \\
D &\to cS_2 \mid cS_5
\end{aligned}
$$

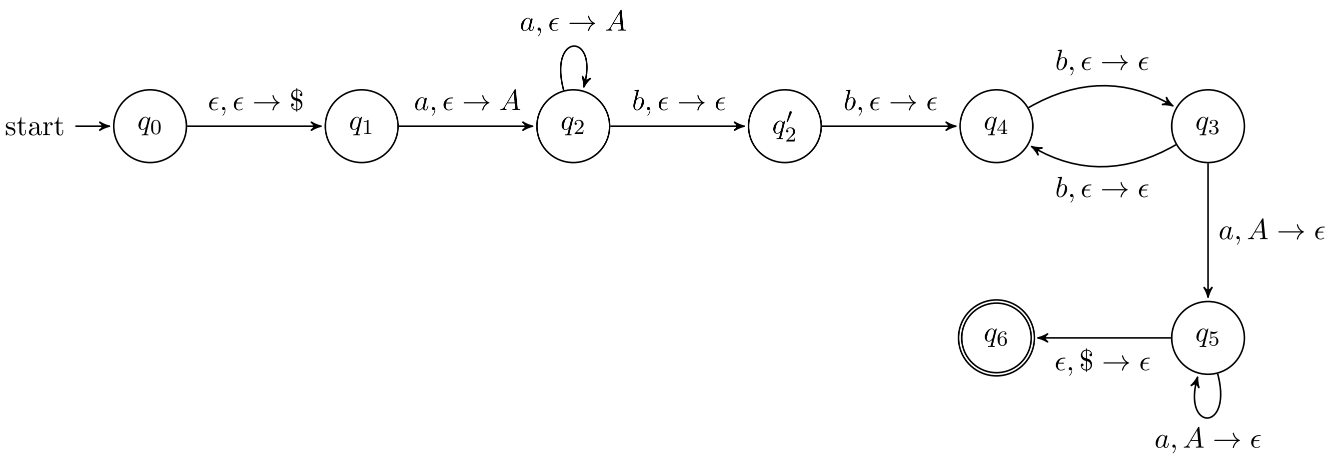
Finally, add $U_3 \to c$ and adjust.

$$
\begin{aligned}
S_0 &\to \varepsilon \mid U_1S_1 \mid U_1U_2 \mid U_3S_2 \mid U_3S_5 \\
S &\to U_1S_1 \mid U_1U_2 \mid U_3S_2 \mid U_3S_5 \\
S_1 &\to SU_2 \\
S_2 &\to U_3S_3 \\
S_3 &\to DS_4 \\
S_4 &\to U_3U_3 \\
S_5 &\to U_3S_6 \\
S_6 &\to U_3U_3 \\
U_1 &\to a \\
U_2 &\to b \\
U_3 &\to c \\
D &\to U_3S_2 \mid U_3S_5
\end{aligned}
$$

Exercise 3: Constructing Pushdown Automata *(4 Points)*

Consider the language $L = \{a^n b^{2m} b a^n \mid m, n > 0\}$ over the alphabet $\Sigma = \{a, b\}$.
Construct a PDA $\mathcal{A}$ with $L(\mathcal{A}) = L$.

Sample Solution



The formal definition of the automaton is implicitly given.

Exercise 4: Pumping Lemma for Context-Free Languages *(3+3 Points)*

Use the pumping lemma to show that the following languages over the alphabet $\Sigma = \{a, b\}$ are not context free:

(a) $\{ww \mid w \in \{a, b\}^*\}$

(b) $\{a^n b a^{2n} b a^{3n} \mid n \geq 0\}$

Sample Solution

(a) Assume the language was context free. Let p be the pumping length. We show that the string $s = a^p b^p a^p b^p$ cannot be pumped, leading to a contradiction. Let $s = uvxyz$ with $|vxy| \leq p$ and $|vy| > 0$.

First, we show that the substring vxy straddles the midpoint of s. If not, then vxy is either fully contained in the first or fully contained in the second half of s. If it is contained in the first half, we obtain that $uv^2xy^2z = tb^p a^p b^p$. Because of $|vy| > 0$ it follows that $|t| > p$ and because of $|uvxy| \leq 2p$ it follows that $|t| < 3p$. So uv^2xy^2z has a b in the first position of its second half, making it impossible to have the form ww. Similarly, if vxy is contained in the second half of s, then the string uv^2xy^2z has an a in the last position of its first half, making it again impossible to have the form ww.

But if vxy straddles the midpoint of s, then because of $|vxy| \leq p$, pumping s down to uxz leads to the string $a^p b^i a^j b^p$. As $|vy| > 0$, either i or j (or both) are strictly less than p. So this string has not the form ww.

(b) Assume the language was context free with p the pumping length. Define $s := a^p b a^{2p} b a^{3p}$ and let $s = uvxyz$ be a decomposition of s with $|vxy| \leq p$ and $|vy| > 0$. We show that uv^2xy^2z cannot be in the language, giving a contradiction. If v or y contained b, the string uv^2xy^2z would have more than two b's and is therefore not in the language. So assume that neither v nor y contains b. That means that v as well as y is fully contained in one of the three segments a^p, a^{2p} and a^{3p}. But then pumping s up to uv^2xy^2z would violate the $1 : 2 : 3$ length ratio of the segments, because the length of at least one segment is changed (as $|vy| > 0$) and at least one segment keeps its length.