## CS21004 Formal Languages and Automata Theory, Spring 2019–2020

### Unrestricted Grammars

---

**1.** Design unrestricted grammars for the following languages.

**(a)** $\{w \in \{a,b,c\}^* \mid \#a(w) = \#b(w) = \#c(w)\}$.

*Solution* We first generate $(ABC)^n$. We then allow $A, B, C$'s to line up in any fashion. Finally, we change them to $a, b, c$, respectively.

$$\begin{aligned} S &\to \varepsilon \mid ABCS \\ AB &\to BA \\ BA &\to AB \\ BC &\to CB \\ CB &\to BC \\ CA &\to AC \\ AC &\to CA \\ A &\to a \\ B &\to b \\ C &\to c \end{aligned}$$

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

*Solution* The start symbol $S$ generates a string $(aA + bB)^*$ and then converts to $T$ at the end. We then allow $A$ and $B$ to cross $a$ and $b$ (but not themselves) to come in contact with $T$ and get converted to $a$ and $b$, respectively.

$$\begin{aligned} S &\to aAS \mid bBS \mid T \\ Aa &\to aA \\ Ab &\to bA \\ AT &\to Ta \\ Ba &\to aB \\ Bb &\to bB \\ BT &\to Tb \\ T &\to \varepsilon \end{aligned}$$

**(c)** $\{a^i b^j c^k d^l \mid i = k \text{ and } j = l\}$.

*Solution* First generate $a^i C^i b^j T d^j$. Then allow the $C$'s to cross the $b$'s, come in contact with $T$, and convert to $c$.

$$\begin{aligned} S &\to UV \\ U &\to \varepsilon \mid aUC \\ V &\to T \mid bVd \\ Cb &\to bC \\ CT &\to Tc \\ T &\to \varepsilon \end{aligned}$$

**2.** Consider the unrestricted grammar over the singleton alphabet $\Sigma = \{a\}$, having the start symbol $S$, and with the following productions.

$$\begin{aligned} S &\to AS \mid aT \\ Aa &\to aaaA \\ AT &\to T \\ T &\to \varepsilon \end{aligned}$$

What is the language generated by this unrestricted grammar? Justify.

*Solution* We have $\mathscr{L}(S) = \{a^{3^n} \mid n \geqslant 0\}$. In order to prove this, we may proceed by induction on the number of $A$'s generated before the rule $S \to aT$ is applied. Each generated $A$ must get in contact with $T$ for vanishing. In the rightward journey of each $A$, the number of $a$'s is tripled.

**3.** Prove that any grammar can be converted to an equivalent grammar with rules of the form $\alpha A \gamma \to \alpha\beta\gamma$ for $A \in N$ and $\alpha, \beta, \gamma \in (\Sigma \cup N)^*$.

*Solution* First, we introduce a non-terminal symbol $T_a$ for each terminal symbol $a$, and add the rule $T_a \to a$. Let us now look at a general rule $\alpha \to \beta$ with $|\alpha| = m$ and $|\beta| = n$. If $\alpha$ and $\beta$ contain terminal symbols, replace them by the corresponding non-terminal symbols introduced above. We can now assume that $\alpha, \beta \in N^*$. In particular, we can write the rule $\alpha \to \beta$ as $U_1U_2 \ldots U_m \to V_1V_2 \ldots V_n$, where the $U_i$ and $V_j$ are all non-terminal symbols. By introducing new non-terminal symbols $W_1, W_2, \ldots, W_m$, we can replace the given rule by a sequence of rules, each of the form given in the question. Notice that $m \geqslant 1$, so the following rules work only in the presence of the new terminal symbols $W_i$ and consequently do not interfere with the existing grammar.

**Case 1:** $m \leqslant n$.

$$
\begin{aligned}
U_1U_2U_3\ldots U_{m-1}U_m &\to W_1U_2U_3\ldots U_{m-1}U_m \\
W_1U_2U_3\ldots U_{m-1}U_m &\to W_1W_2U_3\ldots U_{m-1}U_m \\
W_1W_2U_3\ldots U_{m-1}U_m &\to W_1W_2W_3\ldots U_{m-1}U_m \\
&\ \vdots \\
W_1W_2W_3\ldots U_{m-1}U_m &\to W_1W_2W_3\ldots W_{m-1}U_m \\
W_1W_2W_3\ldots W_{m-1}U_m &\to W_1W_2W_3\ldots W_{m-1}W_mV_{m+1}V_{m+2}\ldots V_n \\
W_1W_2W_3\ldots W_{m-1}W_mV_{m+1}V_{m+2}\ldots V_n &\to V_1W_2W_3\ldots W_{m-1}W_mV_{m+1}V_{m+2}\ldots V_n \\
V_1W_2W_3\ldots W_{m-1}W_mV_{m+1}V_{m+2}\ldots V_n &\to V_1V_2W_3\ldots W_{m-1}W_mV_{m+1}V_{m+2}\ldots V_n \\
V_1V_2W_3\ldots W_{m-1}W_mV_{m+1}V_{m+2}\ldots V_n &\to V_1V_2V_3\ldots W_{m-1}W_mV_{m+1}V_{m+2}\ldots V_n \\
&\ \vdots \\
V_1V_2V_3\ldots W_{m-1}W_nV_{m+1}V_{m+2}\ldots V_n &\to V_1V_2V_3\ldots V_{m-1}W_mV_{m+1}V_{m+2}\ldots V_n \\
V_1V_2V_3\ldots V_{m-1}W_mV_{m+1}V_{m+2}\ldots V_n &\to V_1V_2V_3\ldots V_{m-1}V_mV_{m+1}V_{m+2}\ldots V_n
\end{aligned}
$$

**Case 2:** $m \geqslant n$.

$$
\begin{aligned}
U_1U_2U_3\ldots U_m &\to W_1U_2U_3\ldots U_m \\
W_1U_2U_3\ldots U_m &\to W_1W_2U_3\ldots U_m \\
W_1W_2U_3\ldots U_m &\to W_1W_2W_3\ldots U_m \\
&\ \vdots \\
W_1W_2W_3\ldots W_{m-1}U_m &\to W_1W_2W_3\ldots W_{m-1}W_m \\
W_1W_2W_3\ldots W_nW_{n+1}W_{n+2}\ldots W_m &\to W_1W_2W_3\ldots W_nW_{n+2}\ldots W_m \\
W_1W_2W_3\ldots W_nW_{n+2}W_{n+3}\ldots W_m &\to W_1W_2W_3\ldots W_nW_{n+3}\ldots W_m \\
&\ \vdots \\
W_1W_2W_3\ldots W_nW_m &\to W_1W_2W_3\ldots W_n \\
W_1W_2W_3\ldots W_n &\to V_1W_2W_3\ldots W_n \\
V_1W_2W_3\ldots W_n &\to V_1V_2W_3\ldots W_n \\
V_1V_2W_3\ldots W_n &\to V_1V_2V_3\ldots W_n \\
&\ \vdots \\
V_1V_2V_3\ldots V_{n-1}W_n &\to V_1V_2V_3\ldots V_{n-1}V_n
\end{aligned}
$$

**4.** Write a context-sensitive grammar for the language

$$\{a^nb^nc^n \mid n \geqslant 1\}.$$

*Solution* Each rule in a context-sensitive grammar is of the form $\alpha A \gamma \to \alpha\beta\gamma$ with $|\beta| \geqslant 1$. In particular, rules of the form $A \to \varepsilon$ are not allowed, and so $\varepsilon$ cannot be in the language of a CSG. In view of Exercise 3, however, we can convert arbitrary rules $\alpha \to \beta$ with $|\beta| \geqslant |\alpha|$ to rules of the desired form.

An unrestricted grammar for the same language (with $n \geqslant 0$) is given in the slides. We have to get rid of the terminal symbols $U$ and $T$ which can vanish. Moreover, we have to replace $Cb \to bC$ because this is not of the desired format.

Eliminating $U$ is easy. Use the rules

$$S \quad \to \quad aBC \mid aSBC.$$

Next, let us handle the swap of $B$ and $C$. Add the rules

$$\begin{aligned} CB &\to UB, \\ UB &\to UV, \\ UV &\to BV, \\ BV &\to BC. \end{aligned}$$

When the $B$'s and the $C$'s are properly lined up, $C$ can change to $c$:

$$C \quad \to \quad c.$$

At this point, sentential forms are $a^n B^n c^n$. In order to convert the $B$'s to $b$'s, we use the final two rules:

$$\begin{aligned} aB &\to ab, \\ bB &\to bb. \end{aligned}$$

**Remark:** The rule $B \to b$ cannot be used. Why?