Homework 6

2.1 Recall the CFG G_4 that we gave in Example 2.4. For convenience, let's rename its variables with single letters as follows.

$$\begin{aligned} E &\to E \texttt{+} T \mid T \\ T &\to T \texttt{×} F \mid F \\ F &\to \texttt{(}E\texttt{)} \mid \texttt{a} \end{aligned}$$

Give parse trees and derivations for each string.

a. `a`
b. `a+a`
c. `a+a+a`
d. `((a))`

2.6 Give context-free grammars generating the following languages.

d. $\{x_1\texttt{\#}x_2\texttt{\#}\cdots\texttt{\#}x_k \mid k \geq 1$, each $x_i \in \{\texttt{a},\texttt{b}\}^*$, and for some i and j, $x_i = x_j^{\mathcal{R}}\}$

***2.19** Let CFG G be the following grammar.

$$\begin{aligned} S &\to \texttt{a}S\texttt{b} \mid \texttt{b}Y \mid Y\texttt{a} \\ Y &\to \texttt{b}Y \mid \texttt{a}Y \mid \varepsilon \end{aligned}$$

Give a simple description of $L(G)$ in English. Use that description to give a CFG for $\overline{L(G)}$, the complement of $L(G)$.

***2.27** Let $G = (V, \Sigma, R, \langle \text{STMT} \rangle)$ be the following grammar.

$$\begin{aligned} \langle \text{STMT} \rangle &\to \langle \text{ASSIGN} \rangle \mid \langle \text{IF-THEN} \rangle \mid \langle \text{IF-THEN-ELSE} \rangle \\ \langle \text{IF-THEN} \rangle &\to \texttt{if condition then } \langle \text{STMT} \rangle \\ \langle \text{IF-THEN-ELSE} \rangle &\to \texttt{if condition then } \langle \text{STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\ \langle \text{ASSIGN} \rangle &\to \texttt{a:=1} \end{aligned}$$

$$\Sigma = \{\texttt{if}, \texttt{condition}, \texttt{then}, \texttt{else}, \texttt{a:=1}\}$$
$$V = \{\langle \text{STMT} \rangle, \langle \text{IF-THEN} \rangle, \langle \text{IF-THEN-ELSE} \rangle, \langle \text{ASSIGN} \rangle\}$$

G is a natural-looking grammar for a fragment of a programming language, but G is ambiguous.

a. Show that G is ambiguous.
b. Give a new unambiguous grammar for the same language.

***2.28** Give unambiguous CFGs for the following languages.

c. $\{w \mid$ the number of `a`'s is at least the number of `b`'s in $w\}$

6 Give a rigorous proof that the grammar $S \to aSbS \mid bSaS \mid \varepsilon$ generates every string with equally many a's and b's.

7 Prove that every regular language has a context-free grammar. *Hint:* given a DFA, explain how to transform it into an equivalent grammar.