Grammars

The Big Picture

- a *grammar* defines the syntax of a language
- a *generative grammar* is a set of rules that can be used to generate all the legal strings in a language
- *parsing* a string means determining how the rules could generate that string
- what does this have to do with regular expressions and regular languages?
  - a regular expression describes how to generate all the legal strings in a regular language
  - a DFA provides a process for determining whether a particular regular expression generates that string
  - every regular expression can be specified by a type of grammar
  - more generally, grammars allow the specification of non-regular languages

Rewriting Rules

- a *rewriting rule* (or *production rule* or *production*) $w \rightarrow u$ specifies that the string w can be replaced by the string u

```
aba → cc

abbabac
abbccc
```


Context-Free Grammars

- in a *context-free grammar*, every production has the form $A \rightarrow w$ where A is a single symbol and w is a string of 0 or more symbols
  - "context-free" refers to the idea that A can be replaced anywhere it occurs – there's no dependence on the symbols around A
- the symbols on the left side of production rules are *non-terminal symbols*
  - convention is to denote them with capital letters
  - one, often denoted S, is the *start symbol*
- the other symbols are *terminal symbols*
  - "terminal" because they cannot be further substituted
  - convention is to denote them with lowercase letters

$$
\begin{aligned}
A &\longrightarrow aAbB \\
S &\longrightarrow SS \\
C &\longrightarrow Acc \\
B &\longrightarrow b \\
A &\longrightarrow \varepsilon
\end{aligned}
$$

Context-Free Grammars

- to generate the strings in the language, start with the start symbol and apply production rules
  - the strings in the language are those with only terminal symbols

3. Identify the language generated by each of the following context-free grammars.

a) $S \longrightarrow aaSb$
$S \longrightarrow \varepsilon$

b) $S \longrightarrow aSb$
$S \longrightarrow aaSb$
$S \longrightarrow \varepsilon$

c) $S \longrightarrow TS$
$S \longrightarrow \varepsilon$
$T \longrightarrow aTb$
$T \longrightarrow \varepsilon$

d) $S \longrightarrow ABA$
$A \longrightarrow aA$
$A \longrightarrow a$
$B \longrightarrow bB$
$B \longrightarrow cB$
$B \longrightarrow \varepsilon$

Context-Free Grammars

Definition 4.1. A ***context-free grammar*** is a 4-tuple (V, Σ, P, S), where:

1. V is a finite set of symbols. The elements of V are the non-terminal symbols of the grammar.
2. Σ is a finite set of symbols such that $V \cap \Sigma = \emptyset$. The elements of Σ are the terminal symbols of the grammar.
3. P is a set of production rules. Each rule is of the form $A \longrightarrow w$ where A is one of the symbols in V and w is a string in the language $(V \cup \Sigma)^*$.
4. $S \in V$. S is the start symbol of the grammar.

- informally, a context-free grammar is often specified just by listing the production rules
  - terminal symbols are those on the left side of productions
  - start symbol is the symbol on the left side of the first production listed
  - non-terminal symbols are those that only appear on the right side of productions

More Notation

- for a context-free grammar G,
  - $x \Rightarrow_G y$ denotes that y can be obtained from x by applying one of G's productions i.e. there's a rule $A \rightarrow w$, $x = uAv$, and $y = uwv$
    - read as "x yields y" or "x produces y"
  - $x \Rightarrow^*_G y$ denotes that y can be obtained from x by applying a sequence of 0 or more of G's production rules
    - read as "x yields y in zero or more steps" or "x produces y in zero or more steps"
  - note that often these are written as just $x \Rightarrow y$ and $x \Rightarrow^* y$ (without the G subscript) if the grammar in question is understood

Theorem 4.1. *Let G be the context-free grammar (V, Σ, P, S). Then:*

1. *If x and y are strings in $(V \cup \Sigma)^*$ such that $x \Longrightarrow y$, then $x \Longrightarrow^* y$.*
2. *If x, y, and z are strings in $(V \cup \Sigma)^*$ such that $x \Longrightarrow^* y$ and $y \Longrightarrow^* z$, then $x \Longrightarrow^* z$.*
3. *If x and y are strings in $(V \cup \Sigma)^*$ such that $x \Longrightarrow y$, and if s and t are any strings in $(V \cup \Sigma)^*$, then $sxt \Longrightarrow syt$.*
4. *If x and y are strings in $(V \cup \Sigma)^*$ such that $x \Longrightarrow^* y$, and if s and t are any strings in $(V \cup \Sigma)^*$, then $sxt \Longrightarrow^* syt$.*

Context-Free Languages

Definition 4.2. Suppose that $G = (V, \Sigma, P, S)$ is a context-free grammar. Then the language generated by G is the language $L(G)$ over the alphabet Σ defined by

$$L(G) = \{w \in \Sigma^* \mid S \Longrightarrow^*_G w\}$$

- a language L is a *context-free language* if there is a context-free grammar G such that $L(G) = L$
  - there is not necessarily a unique G producing a given language
  - two context-free grammars that generate the same language are *equivalent*

- the sequence $S \Rightarrow x1 \Rightarrow x2 \Rightarrow \ldots \Rightarrow w$ is a *derivation* of w in the grammar G
  - there may be more than one way to derive a given string

4. For each of the following languages find a context-free grammar that generates the language:

a) $\{a^n b^m \mid n \geq m > 0\}$

b) $\{a^n b^m \mid n, m \in \mathbb{N}\}$

c) $\{a^n b^m \mid n \geq 0 \wedge m = n + 1\}$

d) $\{a^n b^m c^n \mid n, m \in \mathbb{N}\}$

e) $\{a^n b^m c^k \mid n = m + k\}$

f) $\{a^n b^m \mid n \neq m\}$

g) $\{a^n b^m c^r d^t \mid n + m = r + t\}$

h) $\{a^n b^m c^k \mid n \neq m + k\}$