Regular Grammars

Chapter 7
Regular Grammars

A regular grammar $G$ is a quadruple $(V, \Sigma, R, S)$, where:

- $V$ is the rule alphabet, which contains nonterminals and terminals,
- $\Sigma$ (the set of terminals) is a subset of $V$,
- $R$ (the set of rules) is a finite set of rules of the form: $X \rightarrow Y$,
- $S$ (the start symbol) is a nonterminal.
Regular Grammars

In a regular grammar, all rules in $R$ must:

- have a left hand side that is a single nonterminal

- have a right hand side that is:
  - $\varepsilon$, or
  - a single terminal, or
  - a single terminal followed by a single nonterminal.

Legal: $S \rightarrow a$, $S \rightarrow \varepsilon$, and $T \rightarrow aS$

Not legal: $S \rightarrow aSa$ and $aSa \rightarrow T$
Regular Grammar Example

$L = \{ w \in \{a, b\}^* : |w| \text{ is even} \} \quad ((aa) \cup (ab) \cup (ba) \cup (bb))^*$
Regular Grammar Example

\[ L = \{ w \in \{a, b\}^* : |w| \text{ is even} \} \quad ((aa) \cup (ab) \cup (ba) \cup (bb))^* \]

\[
\begin{align*}
S & \rightarrow \varepsilon \\
S & \rightarrow a T \\
S & \rightarrow b T \\
T & \rightarrow a \\
T & \rightarrow b \\
T & \rightarrow a S \\
T & \rightarrow b S
\end{align*}
\]
Regular Languages and Regular Grammars

**Theorem:** The class of languages that can be defined with regular grammars is exactly the regular languages.

**Proof:** By two constructions.
Regular Languages and Regular Grammars

**Regular grammar → FSM:**

\[ \text{grammartosm}(G = (V, \Sigma, R, S)) = \]

1. Create in \( M \) a separate state for each nonterminal in \( V \).
2. Start state is the state corresponding to \( S \).
3. If there are any rules in \( R \) of the form \( X \rightarrow w \), for some \( w \in \Sigma \), create a new state labeled \#. 
4. For each rule of the form \( X \rightarrow w Y \), add a transition from \( X \) to \( Y \) labeled \( w \).
5. For each rule of the form \( X \rightarrow w \), add a transition from \( X \) to \# labeled \( w \).
6. For each rule of the form \( X \rightarrow \varepsilon \), mark state \( X \) as accepting.
7. Mark state \# as accepting.

**FSM → Regular grammar:** Similarly.
Example 1 - Even Length Strings

\[
\begin{align*}
S & \rightarrow \varepsilon \\
S & \rightarrow aT \\
S & \rightarrow bT \\
T & \rightarrow a \\
T & \rightarrow b \\
T & \rightarrow aS \\
T & \rightarrow bS
\end{align*}
\]
Strings that End with $aaaa$

$L = \{w \in \{a, b\}^* : w$ ends with the pattern $aaaa\}$.

\[
\begin{align*}
S & \rightarrow aS \\
S & \rightarrow bS \\
S & \rightarrow aB \\
B & \rightarrow aC \\
C & \rightarrow aD \\
D & \rightarrow a
\end{align*}
\]
Strings that End with aaaa

$L = \{ w \in \{a, b\}^* : w \text{ ends with the pattern aaaa} \}.$

\[
\begin{align*}
S & \rightarrow aS \\
S & \rightarrow bS \\
S & \rightarrow aB \\
B & \rightarrow aC \\
C & \rightarrow aD \\
D & \rightarrow a
\end{align*}
\]
Example 2 – One Character Missing

$S \rightarrow \varepsilon$
$S \rightarrow aB$
$S \rightarrow aC$
$S \rightarrow bA$
$S \rightarrow bC$
$S \rightarrow cA$
$S \rightarrow cB$

$A \rightarrow bA$
$A \rightarrow cA$
$A \rightarrow \varepsilon$
$B \rightarrow aB$
$B \rightarrow cB$
$B \rightarrow \varepsilon$

$C \rightarrow aC$
$C \rightarrow bC$
$C \rightarrow \varepsilon$