Pharos University in Alexandria
Faculty of Computer Science & Artificial Intelligence
Course Title: Theory of Computation
Code: CS 307

Theory of Computation

Lecturer: Sherine Shawky

Text Books

1. Introduction to formal languages and automata, Peter Linz, 6th edition, 2017.

Week 5

Regular Grammars

Regular Grammars (RG)

- A third way of describing regular languages is by means of certain grammars.
- Whenever we define a language family through an automaton or in some other way, we are interested in knowing what kind of grammar we can associate with the family.

Grammars

- **Grammars:**

A grammar G is defined as a quadruple

$$G = (V, T, S, P),$$

- **<u>Definition:</u>**
  - **V** = Set of Variables (non-empty)
  - **T** = Set of Terminal Symbols
  - **S** = Start Symbol
  - **P** = Set of Productions

Grammars

- Example:

Consider the grammar: $G = (\{S\}, \{a, b\}, S, P)$,

with P given by

$$S \rightarrow aSb,$$
$$S \rightarrow \lambda.$$

It generates the language:

$L(G) = \{\lambda, \text{ab}, \text{aabb}, \text{aaabbb}, \ldots\}$

$$L(G) = \{a^n b^n : n \geq 0\},$$

Types of Grammars

Right-Linear Grammars

- A grammar $G = (V, T, S, P)$ is said to be right-linear if all productions are of the form $A \rightarrow xB$, $A \rightarrow x$, where $A, B \in V$, and $x \in T^*$.

- **Left-Linear Grammars**
  - A grammar is said to be left-linear if all productions are of the form $A \rightarrow Bx$, or A → x.

Regular Grammars (RG)

- A regular grammar is one that is either right-linear or left-linear.
- Note that in a regular grammar, at most one variable appears on the right side of any production.
- Furthermore, that variable must consistently be either the rightmost or leftmost symbol of the right side of any production.

Regular Grammars (RG)

- Example
- The grammar $G1 = (\{S\}, \{a, b\}, S, P1)$, with $P1$ given as $S \rightarrow abS|a$ is right-linear.
- The grammar $G2 = (\{S, S1, S2\}, \{a, b\}, S, P2)$, with productions $S \rightarrow S1ab$, $S1 \rightarrow S1ab|S2$, $S2 \rightarrow a$, is left-linear.
- Both $G1$ and $G2$ are regular grammars.

Regular Grammars (RG)

- The sequence $S \Rightarrow abS \Rightarrow ababS \Rightarrow ababa$ is a derivation with the grammar $G1 = (\{S\}, \{a, b\}, S, P1)$, with $P1$ given as $S \rightarrow abS|a$ is right-linear.
- From this single instance it is easy to conjecture that $L(G1)$ is the language denoted by the regular expression $r = (ab)^* a$.

Regular Grammars (RG)

- $L\ (G2)$ is the regular language $L\ (aab\ (ab)^*)$ for the grammar

$G2 = (\{S, S1, S2\}, \{a, b\}, S, P2)$, with productions

$S \rightarrow S1ab,$

$S1 \rightarrow S1ab|S2,$

$S2 \rightarrow a.$

Regular Grammars (RG)

- Example

The grammar $G = (\{S, A, B\}, \{a, b\}, S, P)$ with productions $S \rightarrow A$, $A \rightarrow aB|\lambda$, $B \rightarrow Ab$, is not regular.

- Although every production is either in right-linear or left-linear form, the grammar itself is neither right-linear nor leftlinear, and therefore is not regular.
- The grammar is an example of a linear grammar.
- A linear grammar is a grammar in which at most one variable can occur on the right side of any production, without restriction on the position of this variable.
- Clearly, a regular grammar is always linear, but not all linear grammars are regular.

Regular Grammars (RG)

- THEOREM 3.3

Let $G = (V, T, S, P)$ be a right-linear grammar. Then $L(G)$ is a regular language.

- THEOREM 3.5

A language L is regular if and only if there exists a left-linear grammar G such that $L = L(G)$.

- THEOREM

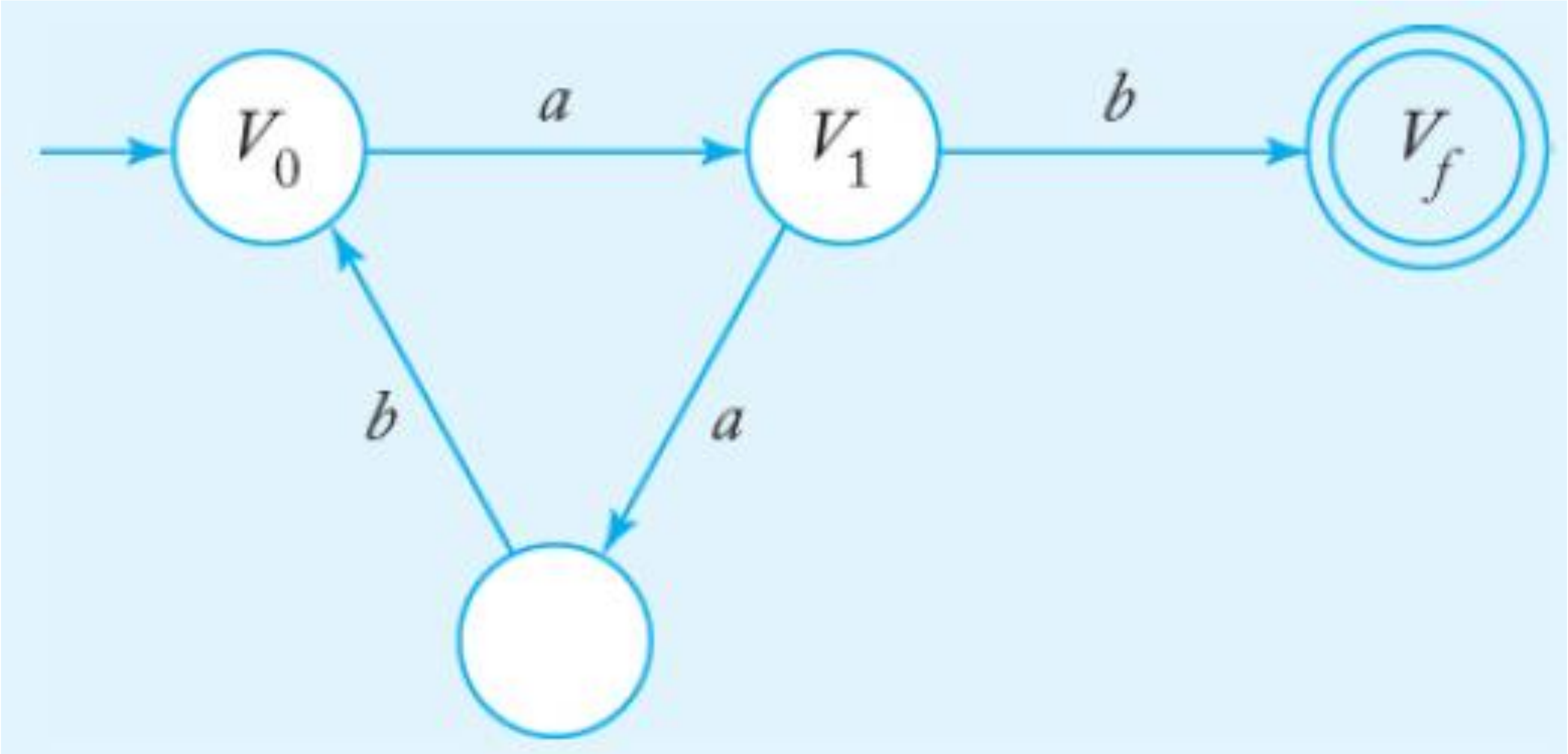
If L is a regular language on the alphabet Σ, then there exists a linear grammar $G = (V, \Sigma, S, P)$ such that $L = L(G)$.

Convert from RG to FA

- EXAMPLE 3.15
- Construct a finite automaton that accepts the language generated by the grammar $V0 \to aV1$, $V1 \to abV0|b$, where $V0$ is the start variable.
- We start the transition graph with vertices $V0$, $V1$, and Vf. The first production rule creates an edge labeled a between $V0$ and $V1$. For the second rule, we need to introduce an additional vertex so that there is a path labeled ab between $V1$ and $V0$.
- Finally, we need to add an edge labeled b between $V1$ and Vf, giving the automaton shown in Figure.
- The language generated by the grammar and accepted by the automaton is the regular language $L((aab)^* ab)$.

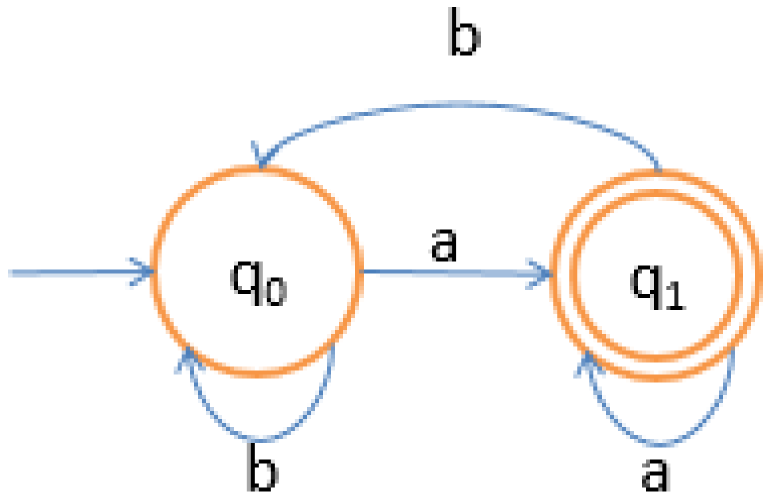
Regular Grammars (RG)

- EXAMPLE 3.15



Convert From FA to RG

- Transform the following DFA to a right regular grammar



Solution:

$Q_0 \rightarrow aQ_1 \mid bQ_0$

$Q_1 \rightarrow aQ_1 \mid bQ_0 \mid \varepsilon$

Self-Study Assignment

Transform from RG to RE and vice versa.