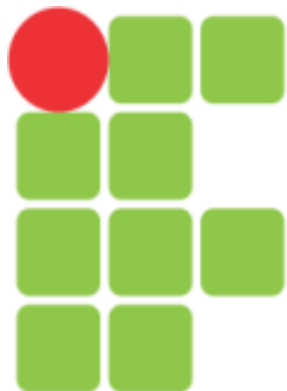




INSTITUTO FEDERAL
CEARÁ



INSTITUTO FEDERAL
CEARÁ



Ernani Andrade Leite

ernani@ifce.edu.br



Gramáticas Regulares

- 1 Introdução e Conceitos Básicos
- 2 Propriedades das Gramáticas Regulares.
 - 2.1 Exemplos.
- 3 Transformação de AFD em Gramáticas Regulares
 - 3.1 Algoritmo de conversão.
 - 3.2 Transformação de Gramáticas Regulares em AFD
 - 3.3 Algoritmo de conversão.
- 4 Exercícios.



Gramáticas Regulares

Gramática Regular (GR). Uma gramática G é regular se todas suas produções são tais que podem ser colocada em alguma das seguintes formas:

- $A \rightarrow a$
- $A \rightarrow aB$
- $A \rightarrow \varepsilon$

Isto é, do lado esquerdo da regra sempre há somente um não-terminal. Do lado direito há um terminal sozinho (acompanhado no máximo de um não-terminal), ou há apenas a palavra vazia (ε).

Às vezes, inicialmente a gramática não possui todas suas regras em alguma das formas segundo a definição de gramática regular. No entanto, fazendo-se pequenas alterações é possível obter uma gramática equivalente onde todas as regras estejam de acordo com a definição de gramática regular. Exemplo: seja G_1 a gramática a seguir:

$$\begin{aligned} S &\rightarrow AB \\ A &\rightarrow aA \mid a \\ B &\rightarrow b \end{aligned}$$



Gramáticas Regulares

6. Transformação de AFD em Gramática Regular (GR)

É possível escrever uma gramática regular para todo AFD. Para tal basta seguir o algoritmo a seguir:

- a cada estado é associado um não-terminal da gramática, sendo o estado inicial q_0 associado ao símbolo inicial (**S**)
- para cada transição de estado representada no grafo cria-se uma regra de produção na gramática, tal que o estado de origem torna-se o não-terminal à esquerda da regra e o estado destino torna-se um não-terminal do lado direita da regra após o terminal lido na transição.
- cria-se uma regra para cada não-terminal associado a um estado final onde o lado direito da regra é formado apenas pela palavra vazia (ϵ).

Gramáticas Regulares

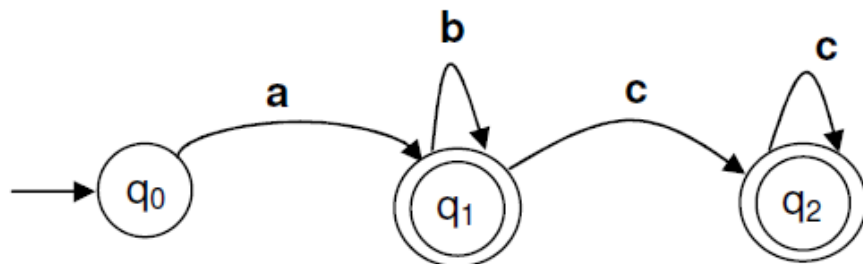


Figura 5

Como exemplo considere o AFD da Figura 5. Fazendo-se $q_0 = S$, $q_1 = A$ e $q_2 = B$, a gramática equivalente ao AFD é:

$$S \rightarrow aA$$

$$A \rightarrow bA \mid cB \mid \varepsilon$$

$$B \rightarrow cB \mid \varepsilon$$

Gramáticas Regulares

7. Transformação de Gramática Regular (GR) em AFD

O algoritmo a seguir mostra como encontrar um AFD equivalente a uma dada gramática regular. Todas as regras da gramática devem ser do tipo $A \rightarrow aB$ (não-terminal levando a um terminal seguido de um único não-terminal) ou $A \rightarrow a$ (não-terminal levando a um terminal sozinho) ou $A \rightarrow \varepsilon$ (não-terminal levando à palavra vazia).

- Para cada não-terminal crie um estado para o AFD, sendo que o símbolo inicial seja o estado inicial ($S = q_0$).
- Para cada regra do tipo $A \rightarrow aB$ crie uma transição que parta do estado A com destino ao estado B através da leitura do terminal a .
- Para cada regra do tipo $A \rightarrow a$ crie uma transição que parta do estado A com destino a um estado A_x , através da leitura do terminal a . Além disso, marque o estado como final
- Para cada regra do tipo $A \rightarrow \varepsilon$, faça do estado associado ao não-terminal A um estado final.

Gramáticas Regulares

A Figura 6 mostra o AFD equivalente à gramática regular a seguir:

$$S \rightarrow aA \mid bB$$

$$A \rightarrow aA \mid bB \mid c \mid \varepsilon$$

$$B \rightarrow bB \mid \varepsilon$$

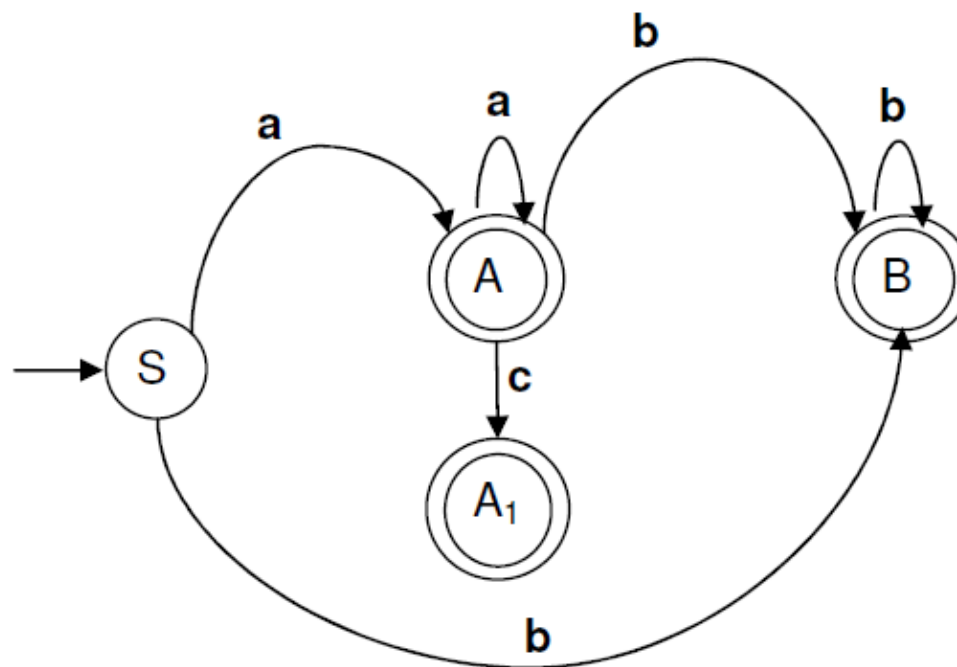


Figura 6

Referências Bibliográficas

- MENEZES, Paulo Blauth. **Linguagens formais e autômatos**. 5.ed. Porto Alegre (RS): Bookman, 2008. 215 p. (Livros Didáticos; v. 3)
- SIPSER, Michael. **Introdução à teoria da computação**. São Paulo (SP): Cengage Learning, 2011. 459 p. Tradução da 2ª edição americana.
- **Fonte:** (material adaptado)
<http://www.icmc.sc.usp.br/~mdgvnune/download/sce5832/Teoria.html>