Aspectos Teóricos da Computação

CIÊNCIAS DA COMPUTAÇÃO

Aspectos Teóricos da Computação

Prof. César C. Xavier

Aspectos Teóricos da Computação

ROTEIRO

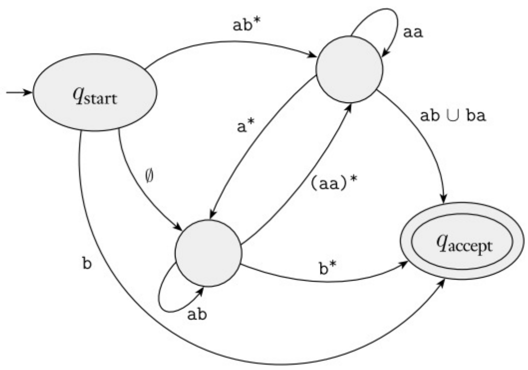
Máquinas de Turing – Parte II

- Autômato Finito Não-Determinístico Generalizado (AFNG - *GNFA*)
- Linguagens Não Regulares
- Gramáticas Livres de Contexto

Aspectos Teóricos da Computação

AFNG

- É todo AFND que aceita como transição qualquer expressão regular como rótulo, ao invés de apenas membros do alfabeto ou ε.



Aspectos Teóricos da Computação

AFNG

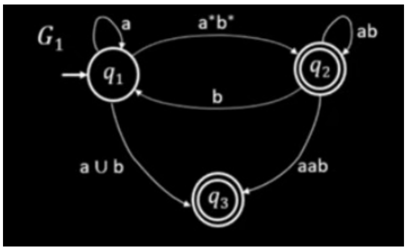
- Condições:

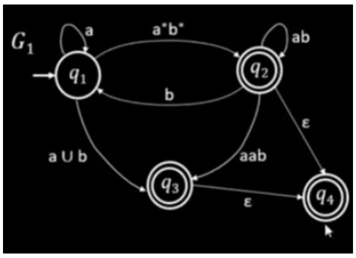
1) O estado inicial tem setas de transição saindo para todos os outros estados, mas nenhuma seta chegando de qualquer outro estado.
2) Existe apenas um estado de aceitação.
3) O estado de aceitação tem setas chegando de todos os outros estados.
4) O estado de aceitação não tem nenhuma seta saindo para qualquer outro estado.
5) O estado de aceitação não é o mesmo do estado inicial.
6) Uma seta sai de cada estado para todos os outros estados e também de cada estado para ele mesmo, com exceção dos estados inicial e de aceitação.

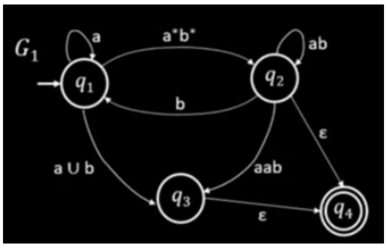
Aspectos Teóricos da Computação

AFNG

- Ex. é possível converter o AFND, a seguir, que possui dois estados finais, para apenas um estado final sem alterar a sua gramática?



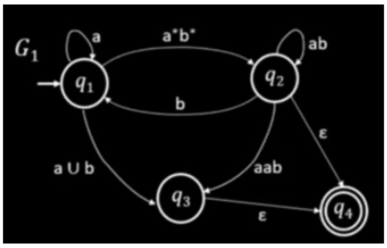


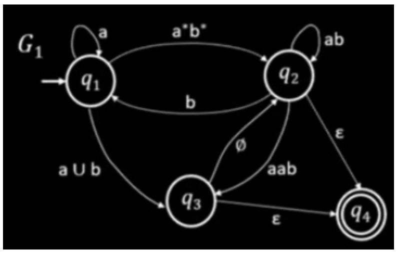


Aspectos Teóricos da Computação

AFNG

- Ex. é possível converter o AFND, a seguir, de forma que existe uma seta que conecte todos os estados finais a um único estado final sem alterar a sua gramática?

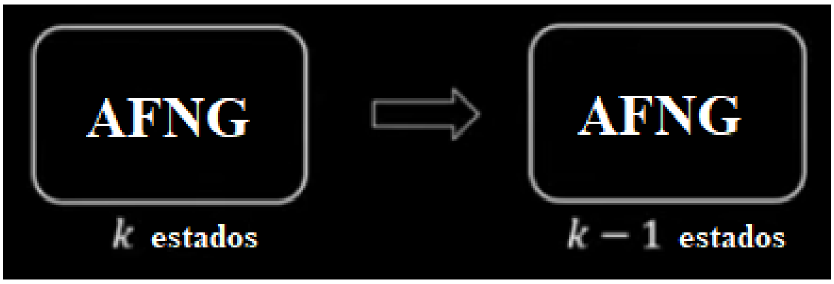




Aspectos Teóricos da Computação

AFNG

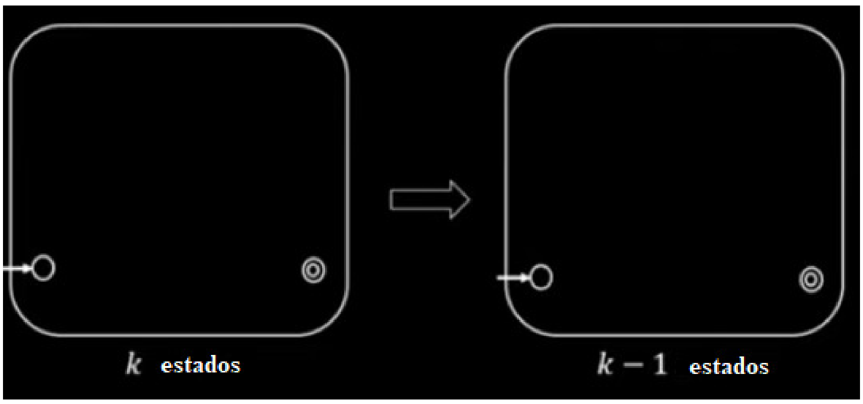
- Ex. é possível converter um AFNG que possuí *k* estados em outro AFNG com *k*-1 estados sem alterar a sua gramática?



Aspectos Teóricos da Computação

AFNG

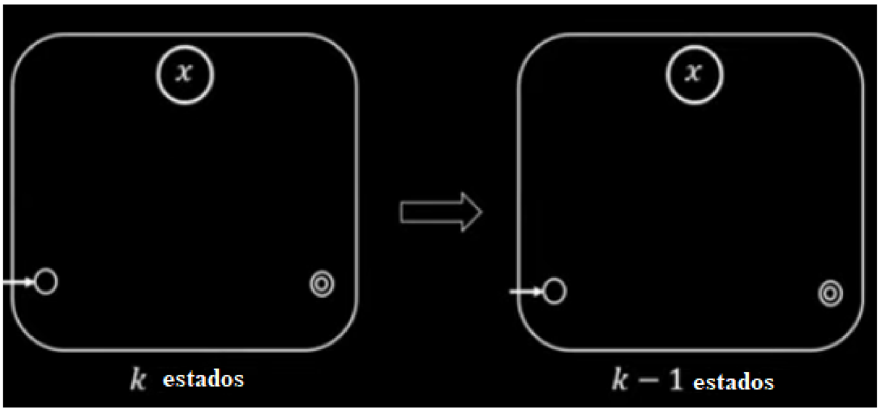
- Ex. é possível converter um AFNG que possuí *k* estados em outro AFNG com *k*-1 estados sem alterar a sua gramática?



Aspectos Teóricos da Computação

AFNG

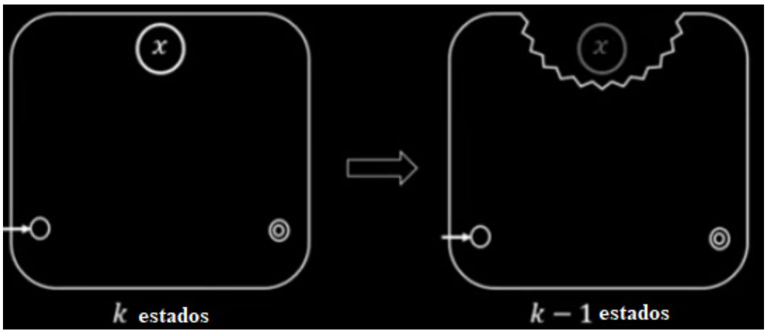
- Ex. é possível converter um AFNG que possuí *k* estados em outro AFNG com *k*-1 estados sem alterar a sua gramática?



Aspectos Teóricos da Computação

AFNG

- Ex. é possível converter um AFNG que possuí *k* estados em outro AFNG com *k*-1 estados sem alterar a sua gramática?



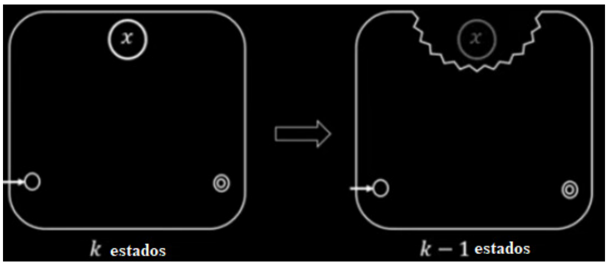
Aspectos Teóricos da Computação

AFNG

- Ex. é possível converter um AFNG que possuí *k* estados em outro AFNG com *k*-1 estados sem alterar a sua gramática?

  1) Escolhemos um estado *x* diferente dos estados inicial e de aceitação.

  2) Removemos *x*

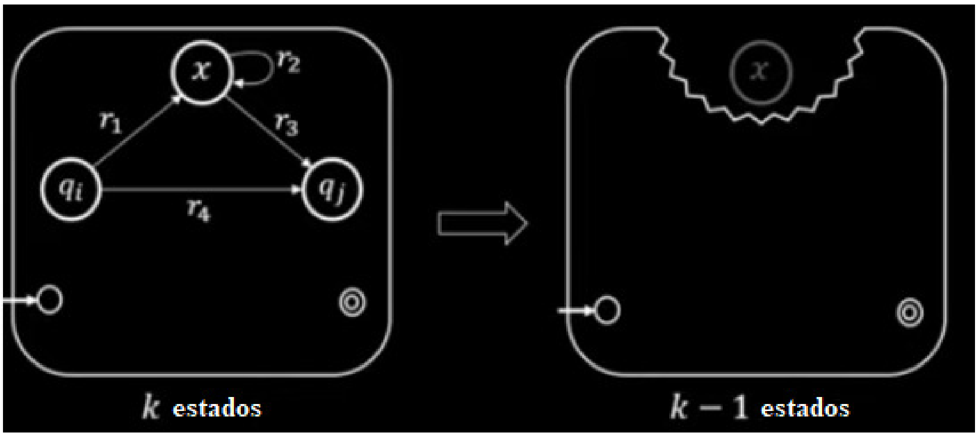


  3) Deve-se reparar o “estrago” devido à remoção dos caminhos que passam por *x*.

Aspectos Teóricos da Computação

AFNG

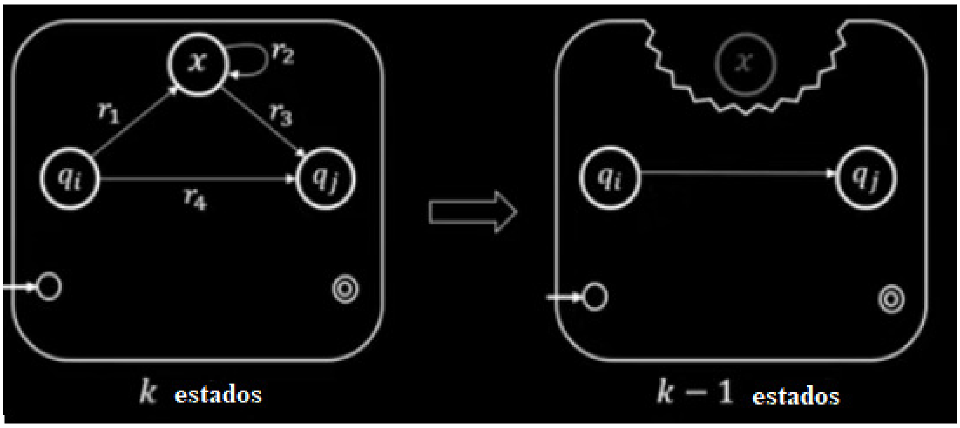
- Ex. é possível converter um AFNG que possuí *k* estados em outro AFNG com *k*-1 estados sem alterar a sua gramática?



Aspectos Teóricos da Computação

AFNG

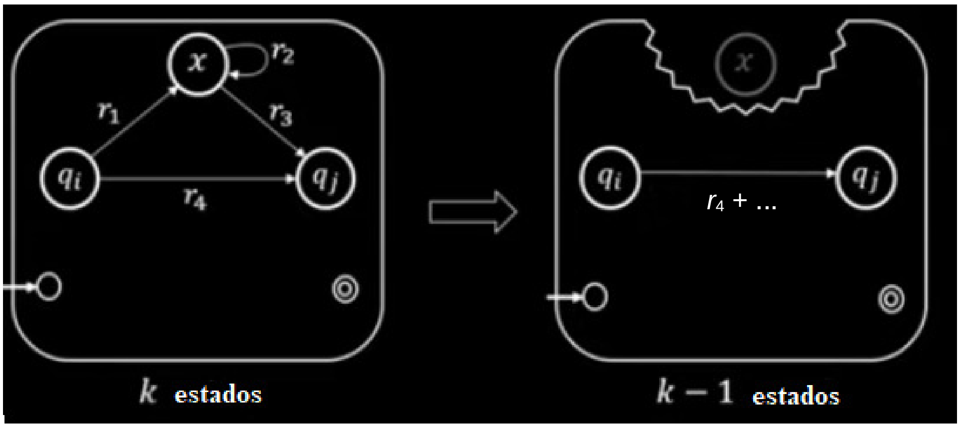
- Ex. é possível converter um AFNG que possuí *k* estados em outro AFNG com *k*-1 estados sem alterar a sua gramática?



Aspectos Teóricos da Computação

AFNG

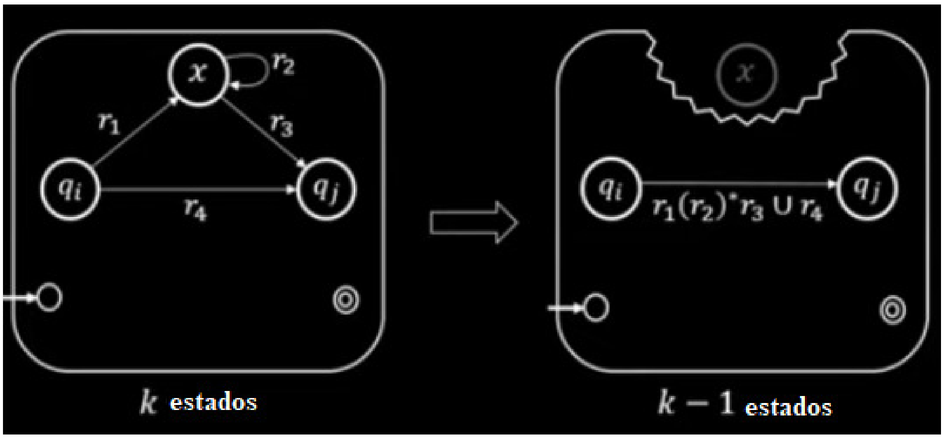
- Ex. é possível converter um AFNG que possuí *k* estados em outro AFNG com *k*-1 estados sem alterar a sua gramática?



Aspectos Teóricos da Computação

AFNG

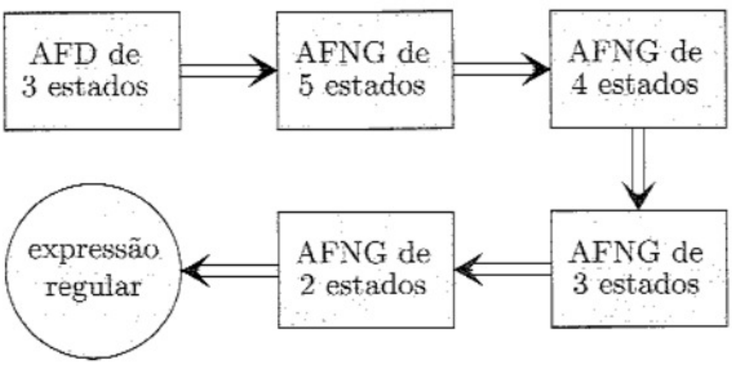
- Ex. é possível converter um AFNG que possuí *k* estados em outro AFNG com *k*-1 estados sem alterar a sua gramática?



Aspectos Teóricos da Computação

AFNG

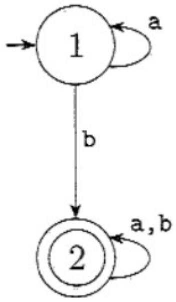
- Etapas para conversão de um AFD de três estados em uma expressão regular:



Aspectos Teóricos da Computação

AFNG

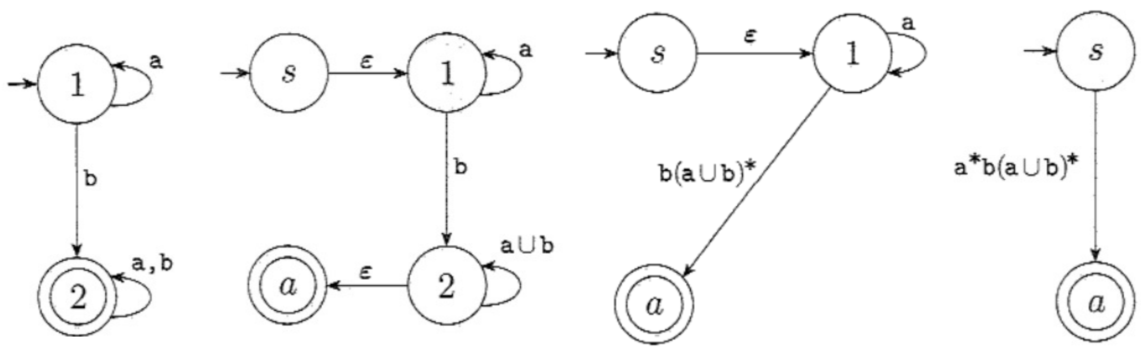
- Determine a expressão regular para o autômato o seguinte autômato:



Aspectos Teóricos da Computação

AFNG

- Determine a expressão regular para o autômato o seguinte autômato:



(b) (c) (d)

Linguagens Não Regulares

- Se uma linguagem é regular existe um AFD.
- Para mostrar que uma linguagem é não regular uma prova deve ser dada.
- Não é suficiente dizer que não foi encontrado um AFD e por isso ela é não regular.
- Exemplos onde $\Sigma = \{0, 1\}$.
  - a) Seja $B = \{ w \mid w$ tenha o mesmo número de 0 e de 1$\}$
    - Intuição: *B* não é regular pois AFD não podem contar infinitamente com um número finito de estados.
  - b) Seja $C = \{ w \mid w$ tenha o mesmo número de 01 e de 10$\}$
    - $0101 \notin C$ e $0110 \in C$.
    - Intuição: *C* não é regular pois AFD não podem contar infinitamente com um número finito de estados, entretanto *C* **é regular**.
    - Algumas vezes a intuição funciona mas pode, ocasionalmente estar errada.

UNIVERSIDADE PAULISTA

Aspectos Teóricos da Computação

Linguagens Não Regulares

Trabalho #1

- Descreva o Lema do Bombeamento e demonstre que as seguintes linguagens não são regulares:
  - $A = \{ 0^n 1^n \mid n \geq 0 \}$; e
  - $B = \{ w \mid w$ tem o mesmo número de 0 e de 1 $\}$

  Prazo: Dia prova NP1 (19:00h)
  E-mail: cesar.xavier@docente.unip.br
  Assunto e-mail: Trabalho bombeamento
  Trabalho: Observar as normas de trabalhos acadêmicos UNIP.

Prof. César C. Xavier

Gramáticas Livres de Contexto

- Método mais poderoso de descrever linguagens.
- Podem ter uma estrutura recursiva.
- Utilizadas inicialmente no estudo de linguagens humanas: nome, verbo, preposição.
- Aplicação em especificação e compilação de linguagens de programação.
- Maioria compiladores e interpretadores possuem um **analisador** que extrai o significado de um programa antes de gerar o código compilado.

UNIP UNIVERSIDADE PAULISTA

Aspectos Teóricos da Computação

Gramáticas Livres de Contexto

- Exemplo de uma gramática livre de contexto G_1:

$$A \rightarrow 0A1$$
$$A \rightarrow B$$
$$B \rightarrow \#$$

- Gramática possui uma coleção de ***regras de substituição***.
- Cada regra aparece em uma linha com um símbolo e uma cadeia separadas por uma seta.
- O símbolo é denominado de ***variável***.
- Variável inicial é aquela que ocorre na primeira regra do lado esquerdo.
- Cadeia é constituída de variáveis e de outros símbolos denominados de ***terminais***.

Prof. César C. Xavier

Gramáticas Livres de Contexto

- Exemplo de uma gramática livre de contexto G_1:

$$A \rightarrow 0A1$$
$$A \rightarrow B$$
$$B \rightarrow \#$$

- G_1 tem quantas regras?
  - 3.
- Quais são as variáveis de G_1?
  - *A* e *B*, onde *A* é a variável inicial.
- Quais são os terminais de G_1?
  - 0, 1 e #.

Gramáticas Livres de Contexto

- Gramáticas geram cadeias de uma linguagem:

  1) Escreva a variável inicial.
  2) Substitua qualquer variável de acordo com uma regra.
  3) Repita o passo 2 até que reste apenas terminais (ou não reste variável).
- Ex.: G_1 gera a cadeia 000#111. A sequencia de substituições para se obter esta cadeia é denominado de ***derivação***.

  G_1:

$$A \rightarrow 0A1$$
$$A \rightarrow B$$
$$B \rightarrow \#$$

Aspectos Teóricos da Computação

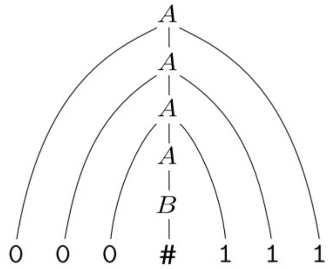
Gramáticas Livres de Contexto

- Ex.: G_1 gera a cadeia 000#111. A sequencia de substituições para se obter esta cadeia é denominado de ***derivação***.
  G_1:

$$A \rightarrow 0A1$$
$$A \rightarrow B$$
$$B \rightarrow \#$$

Derivação: $A \Rightarrow 0A1 \Rightarrow 00A11 \Rightarrow 000A111 \Rightarrow 000B111 \Rightarrow 000\#111.$

Árvore Sintática:



$L(G_1) = \{0^k 1^k \mid k \geq 0\}$

Aspectos Teóricos da Computação

Gramáticas Livres de Contexto

- Seja a gramática livre de contexto G_2:

$$S \rightarrow RR$$
$$R \rightarrow 0R1$$
$$R \rightarrow \varepsilon$$

- Diga se as cadeias a seguir estão em $L(G_2)$:
  - 001101
  - 000111
  - 1010
  - ε