

LFA e Máquina de Turing

Esdras Lins Bispo Jr.
bispojr@ufg.br

Teoria da Computação
Bacharelado em Ciência da Computação

23 de abril de 2014

Plano de Aula

- 1 Pensamento
- 2 Avisos
- 3 Revisão
 - Autômato Finito Não-Determinístico
- 4 LFA
 - Expressões Regulares
- 5 Máquinas de Turing

Sumário

- 1 **Pensamento**
- 2 Avisos
- 3 Revisão
 - Autômato Finito Não-Determinístico
- 4 LFA
 - Expressões Regulares
- 5 Máquinas de Turing

Pensamento



Pensamento



Frase

A moderação e a coragem, portanto, são destruídas pela deficiência e pelo excesso e preservadas pelo meio termo.

Quem?

Aristóteles (384 a.C. - 322 a.C.)
Filósofo e lógico grego.

Sumário

- 1 Pensamento
- 2 **Avisos**
- 3 Revisão
 - Autômato Finito Não-Determinístico
- 4 LFA
 - Expressões Regulares
- 5 Máquinas de Turing

Avisos

Questão Avaliada 02 no Canvas

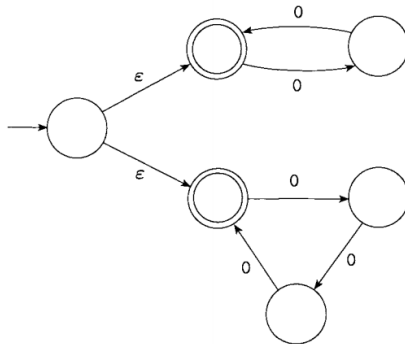
Devo disponibilizá-la novamente!!!

Sumário

- 1 Pensamento
- 2 Avisos
- 3 Revisão**
 - Autômato Finito Não-Determinístico
- 4 LFA
 - Expressões Regulares
- 5 Máquinas de Turing

Autômatos Finitos Não-Determinístico

Qual linguagem este AFN reconhece?



Autômatos Finitos Não-Determinístico

Teorema 1.39

Todo autômato finito não-determinístico tem um autômato finito determinístico equivalente.

Autômatos Finitos Não-Determinístico

Teorema 1.39

Todo autômato finito não-determinístico tem um autômato finito determinístico equivalente.

Corolário 1.40

Uma linguagem é regular se e somente se algum autômato finito não-determinístico a reconhece.

Sumário

- 1 Pensamento
- 2 Avisos
- 3 Revisão
 - Autômato Finito Não-Determinístico
- 4 LFA
 - Expressões Regulares
- 5 Máquinas de Turing

Expressões Regulares

Digamos que R é uma **expressão regular** (ER) se R for:

- ① a , para algum $a \in \Sigma$,
- ② ϵ ,
- ③ \emptyset ,
- ④ $(R_1 \cup R_2)$, em que R_1 e R_2 são expressões regulares,
- ⑤ $(R_1 \circ R_2)$, em que R_1 e R_2 são expressões regulares,
- ⑥ (R_1^*) , em que R_1 é uma expressão regular.

Exemplos de ER

- 0^*10^*
- $\Sigma^*1\Sigma^*$
- $\Sigma^*001\Sigma^*$
- $1^*(01^+)^*$
- $(\Sigma\Sigma)^*$
- $(0 \cup \epsilon)1^* = 01^* \cup 1^*$
- $1^*\emptyset = \emptyset$
- $\emptyset^* = \{\epsilon\}$

Expressões Regulares

Teorema

Uma linguagem é regular se e somente se alguma expressão regular a descreve.

Expressões Regulares

Teorema

Uma linguagem é regular se e somente se alguma expressão regular a descreve.

Estratégia

Utilizar para realizar a prova um **autômato finito não-determinístico generalizado**.

Linguagens Não-Regulares

Existem linguagens que não são regulares como
 $A = \{0^n 1^n \mid n \geq 0\}.$

Linguagens Não-Regulares

Existem linguagens que não são regulares como
 $A = \{0^n 1^n \mid n \geq 0\}.$

Lema do Bombeamento

Se A é uma linguagem regular, então existe um número p (o comprimento do bombeamento) tal que, se s é qualquer cadeia de A de comprimento no mínimo p , então s pode ser dividida em três partes, $s = xyz$, satisfazendo as seguintes condições:

- 1 para cada $i \geq 0$, $xy^i z \in A$,
- 2 $|y| > 0$, e
- 3 $|xy| \leq p$.



Sumário

- 1 Pensamento
- 2 Avisos
- 3 Revisão
 - Autômato Finito Não-Determinístico
- 4 LFA
 - Expressões Regulares
- 5 Máquinas de Turing

Modelos Básicos Computacionais

AFDs, AFNs, e Expressões Regulares

- Potencialidades: reconhecem linguagens como $(10 \cup 1)^*$;
- Fragilidades: não reconhecem linguagens como $A = \{0^n 1^n \mid n \geq 0 \text{ e } n \in \mathbb{N}\}$.

Modelos Básicos Computacionais

AFDs, AFNs, e Expressões Regulares

- Potencialidades: reconhecem linguagens como $(10 \cup 1)^*$;
- Fragilidades: não reconhecem linguagens como $A = \{0^n 1^n \mid n \geq 0 \text{ e } n \in \mathbb{N}\}$.

GLCs e Autômatos com Pilha

- Potencialidades: reconhecem linguagens como $A = \{0^n 1^n \mid n \geq 0 \text{ e } n \in \mathbb{N}\}$;
- Fragilidades: não reconhecem linguagens como $A = \{a^n b^n c^n \mid n \geq 0 \text{ e } n \in \mathbb{N}\}$.

Modelos Básicos Computacionais

AFDs, AFNs, e Expressões Regulares

- Potencialidades: reconhecem linguagens como $(10 \cup 1)^*$;
- Fragilidades: não reconhecem linguagens como $A = \{0^n 1^n \mid n \geq 0 \text{ e } n \in \mathbb{N}\}$.

GLCs e Autômatos com Pilha

- Potencialidades: reconhecem linguagens como $A = \{0^n 1^n \mid n \geq 0 \text{ e } n \in \mathbb{N}\}$;
- Fragilidades: não reconhecem linguagens como $A = \{a^n b^n c^n \mid n \geq 0 \text{ e } n \in \mathbb{N}\}$.

Portanto são bem restritos para servir de modelo de computadores de propósito geral.

Máquinas de Turing (MT)

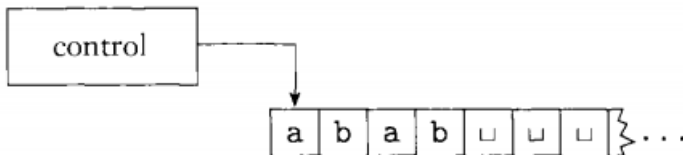
- Modelo mais poderoso que GLCs e AFDs;
- Turing, 1936;
- Características importantes:
 - 1 faz tudo o que um computador real pode fazer;
 - 2 existem certos problemas que uma MT não pode resolver.

Máquinas de Turing (MT)



- *Salaminh salah-mês...* tranforme as figuras em inglês!

Máquinas de Turing (MT)



Máquinas de Turing (MT)

Diferenças entre MT e AFDs

- Uma MT pode tanto escrever sobre a fita quanto ler a partir dela;
- A cabeça de leitura-escrita pode mover-se tanto para a esquerda quanto para a direita;
- A fita é infinita;
- Os estados especiais para rejeitar e aceitar fazem efeito imediatamente.

Máquinas de Turing (MT)

Construindo uma MT

Construir M_1 que reconheça a linguagem

$$B = \{\omega\#\omega \mid \omega \in \{0, 1\}^*\}.$$

LFA e Máquina de Turing

Esdras Lins Bispo Jr.
bispojr@ufg.br

Teoria da Computação
Bacharelado em Ciência da Computação

23 de abril de 2014