

# Decidibilidade

Esdras Lins Bispo Jr.  
bispojr@ufg.br

Teoria da Computação  
Bacharelado em Ciência da Computação

**08 de junho de 2017**

# Plano de Aula

- 1 Revisão
  - Decidibilidade
  
- 2 Decidibilidade
  - O Problema da Parada

# Sumário

- 1 Revisão
  - Decidibilidade
- 2 Decidibilidade
  - O Problema da Parada

# Linguagens Decidíveis

## Exemplos de Linguagens Decidíveis

São úteis porque

- Algumas linguagens decidíveis estão associadas a aplicações;
- Algumas linguagens aparentemente triviais não são decidíveis.

## Problema da aceitação

Dado um modelo computacional  $MC$  e uma cadeia de entrada  $\omega$ , identificar se  $MC$  aceita  $\omega$ .

# Problema da aceitação para AFDs

## Problema da aceitação para AFDs

Dado um AFD  $B$  e uma cadeia de entrada  $\omega$ , identificar se  $B$  aceita  $\omega$ .

## Problema

$A_{AFD} = \{ \langle B, \omega \rangle \mid B \text{ é um AFD que aceita a cadeia de entrada } \omega \}$

## Estratégia de Resolução

Resolver o problema da aceitação para AFDs é decidir se  $\omega \in A_{AFD}$ .

# Problema da aceitação para AFDs

## Teorema 4.1

$A_{AFD}$  é uma linguagem decidível.

## Ideia da Prova

$M =$  “Sobre a entrada  $\langle B, \omega \rangle$ , em que  $B$  é um AFD, e  $\omega$ , uma cadeia:

- 1 Simule  $B$  sobre a entrada  $\omega$ ;
- 2 Se a simulação termina em um estado de aceitação, **aceite**.  
Senão, **rejeite**.”

# Problema da aceitação para AFDs

## Detalhes de implementação

- A entrada  $\langle B, \omega \rangle$  representa um AFD e uma cadeia;
  - Uma representação razoável de  $B$  seria uma lista de seus cinco componentes:  $Q, \Sigma, \delta, q_0$  e  $F$ ;
  - $M$  simula  $B$  de forma que  $M$  **aceita** se  $B$  estiver em um estado final, e **rejeita**, caso contrário.

# Problema da aceitação para AFNs

## Problema da aceitação para AFNs

Dado um AFN  $B$  e uma cadeia de entrada  $\omega$ , identificar se  $B$  aceita  $\omega$ .

## Problema

$A_{AFN} = \{ \langle B, \omega \rangle \mid B \text{ é um AFN que aceita a cadeia de entrada } \omega \}$

## Estratégia de Resolução

Decidir se  $\langle B, \omega \rangle \in A_{AFN}$ .



# Problema da aceitação para AFNs

## Teorema 4.2

$A_{AFN}$  é uma linguagem decidível.

## Prova

$N =$  “Sobre a entrada  $\langle B, \omega \rangle$ , em que  $B$  é um AFN, e  $\omega$ , uma cadeia:

- 1 Converta AFN  $B$  para um AFD equivalente  $C$ , usando o procedimento para essa conversão dado no Teorema 1.39;
- 2 Rode a MT  $M$  do Teorema 4.1 sobre a cadeia  $\langle C, \omega \rangle$ ;
- 3 Se  $M$  aceita, **aceite**. Caso contrário, **rejeite**.”

# Problema da Vacuidade de uma Linguagem

## Descrição

Dada uma linguagem  $L$ , identificar se  $L = \emptyset$ .

## Problema aplicado a AFDs

$$V_{AFD} = \{ \langle A \rangle \mid A \text{ é um AFD e } L(A) = \emptyset \}$$

## Estratégia de Resolução

Decidir se  $\langle A \rangle \in V_{AFD}$ .

# Problema da Vacuidade de uma Linguagem

## Teorema 4.4

$V_{AFD}$  é uma linguagem decidível.

## Prova

A seguinte MT  $T$  decide  $V_{AFD}$ .

$T$  = “Sobre a entrada  $\langle A \rangle$ , em que  $A$  é uma AFD:

- ① Marque o estado inicial de  $A$ ;
- ② Repita até que nenhum estado novo venha a ser marcado;
  - ① Marque qualquer estado que tenha uma transição chegando nele a partir de qualquer estado que já está marcado.
- ③ Se nenhum estado final estiver marcado, **aceite**. Caso contrário, **rejeite**.”

# Sumário

- 1 Revisão
  - Decidibilidade
- 2 Decidibilidade
  - O Problema da Parada

# Problema da Igualdade de Linguagens

## Descrição

Dadas duas linguagem  $L_1$  e  $L_2$ , identificar se  $L_1 = L_2$ .

# Problema da Igualdade de Linguagens

## Descrição

Dadas duas linguagem  $L_1$  e  $L_2$ , identificar se  $L_1 = L_2$ .

## Problema aplicado a AFDs

$EQ_{AFD} = \{\langle A, B \rangle \mid A \text{ e } B \text{ são AFDs e } L(A) = L(B)\}$

# Problema da Igualdade de Linguagens

## Descrição

Dadas duas linguagem  $L_1$  e  $L_2$ , identificar se  $L_1 = L_2$ .

## Problema aplicado a AFDs

$EQ_{AFD} = \{ \langle A, B \rangle \mid A \text{ e } B \text{ são AFDs e } L(A) = L(B) \}$

## Estratégia de Resolução

Decidir se  $\langle A, B \rangle \in EQ_{AFD}$ .

# Problema da Igualdade de Linguagens

## Teorema 4.5

$EQ_{AFD}$  é uma linguagem decidível.



# Problema da Igualdade de Linguagens

## Teorema 4.5

$EQ_{AFD}$  é uma linguagem decidível.

## Lema 4.1

Iremos construir um AFD  $C$  a partir de  $A$  e  $B$  de forma que  $C$  aceite as cadeias que são aceitas por  $A$  ou por  $B$ , mas não por ambas. Consequentemente, se  $A$  e  $B$  reconhecem a mesma linguagem,  $C$  não aceitará nada. A linguagem de  $C$  é

$$L(C) = (L(A) \cap \overline{L(B)}) \cup (\overline{L(A)} \cap L(B))$$

# Problema da Igualdade de Linguagens

## Teorema 4.5

$EQ_{AFD}$  é uma linguagem decidível.

## Lema 4.1

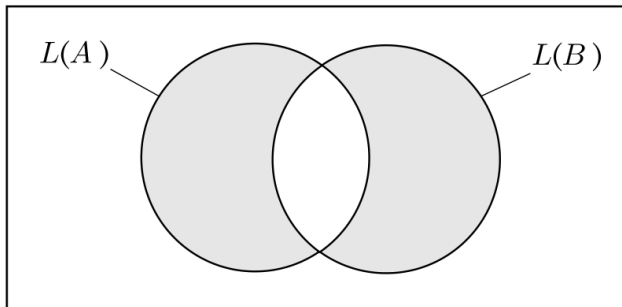
Iremos construir um AFD  $C$  a partir de  $A$  e  $B$  de forma que  $C$  aceite as cadeias que são aceitas por  $A$  ou por  $B$ , mas não por ambas. Consequentemente, se  $A$  e  $B$  reconhecem a mesma linguagem,  $C$  não aceitará nada. A linguagem de  $C$  é

$$L(C) = (L(A) \cap \overline{L(B)}) \cup (\overline{L(A)} \cap L(B))$$

## Corolário

$$L(C) = \emptyset \iff L(A) = L(B)$$

# Problema da Igualdade de Linguagens



**FIGURA 4.6**

A diferença simétrica de  $L(A)$  e  $L(B)$

# Problema da Igualdade de Linguagens

## Teorema 4.5

$EQ_{AFD}$  é uma linguagem decidível.

# Problema da Igualdade de Linguagens

## Teorema 4.5

$EQ_{AFD}$  é uma linguagem decidível.

## Prova

A seguinte MT  $F$  decide  $EQ_{AFD}$ .

$F$  = “Sobre a entrada  $\langle A, B \rangle$ , em que  $A$  e  $B$  são AFDs:

- 1 Construa o AFD  $C$  conforme descrito no Lema 4.1;
- 2 Rode MT  $T$  do Teorema 4.4 sobre a entrada  $\langle C \rangle$ ;
- 3 Se  $T$  aceita, **aceite**. Caso contrário, **rejeite**.”

# Teoremas sobre GLC

## Teorema 4.7

$A_{GLC}$  é uma linguagem decidível.

## Teorema 4.8

$V_{GLC}$  é uma linguagem decidível.

## Teorema 4.9

Toda LLC é decidível.

# Teoremas sobre GLC

## Teorema 4.7

$A_{GLC}$  é uma linguagem decidível.

## Teorema 4.8

$V_{GLC}$  é uma linguagem decidível.

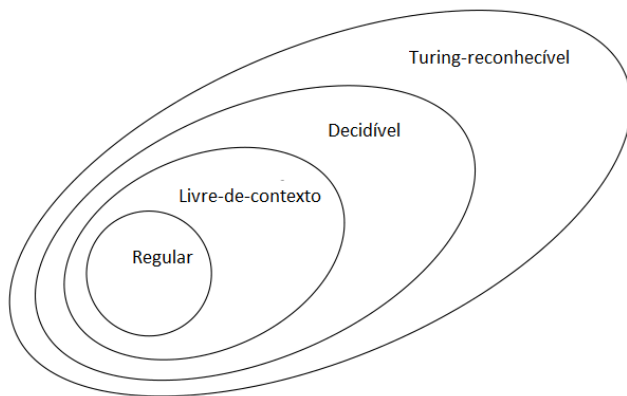
## Teorema 4.9

Toda LLC é decidível.

## Cuidado!

$E_{Q_{GLC}}$  **não** é uma linguagem decidível.

# Relacionamento entre as classes de linguagens



**FIGURA 4.10**

O relacionamento entre classes de linguagens



# O Problema da Parada

## Problema da aceitação para MT

Dada uma MT  $M$  e uma cadeia de entrada  $\omega$ , identificar se  $M$  aceita  $\omega$ .

# O Problema da Parada

## Problema da aceitação para MT

Dada uma MT  $M$  e uma cadeia de entrada  $\omega$ , identificar se  $M$  aceita  $\omega$ .

## Problema

$A_{MT} = \{ \langle M, \omega \rangle \mid M \text{ é uma MT que aceita a cadeia de entrada } \omega \}$

# O Problema da Parada

## Problema da aceitação para MT

Dada uma MT  $M$  e uma cadeia de entrada  $\omega$ , identificar se  $M$  aceita  $\omega$ .

## Problema

$A_{MT} = \{ \langle M, \omega \rangle \mid M \text{ é uma MT que aceita a cadeia de entrada } \omega \}$

## Teorema 4.11

$A_{MT}$  é indecidível.

# O Problema da Parada

## Considerações sobre o Teorema 4.11

$A_{MT}$  é Turing-reconhecível. Pois é possível construir  $U$  da seguinte forma:

# O Problema da Parada

## Considerações sobre o Teorema 4.11

$A_{MT}$  é Turing-reconhecível. Pois é possível construir  $U$  da seguinte forma:

$U =$  “Sobre a entrada  $\langle M, \omega \rangle$ , em que  $M$  é uma MT e  $\omega$  uma cadeia:

- 1 Simule  $M$  sobre a entrada  $\omega$ ;
- 2 Se  $M$  em algum momento entra no seu estado de aceitação, **aceite**; se  $M$  em algum momento entra em seu estado de rejeição, **rejeite**.”

# O Problema da Parada

## Considerações sobre o Teorema 4.11

$A_{MT}$  é Turing-reconhecível. Pois é possível construir  $U$  da seguinte forma:

$U =$  “Sobre a entrada  $\langle M, \omega \rangle$ , em que  $M$  é uma MT e  $\omega$  uma cadeia:

- 1 Simule  $M$  sobre a entrada  $\omega$ ;
- 2 Se  $M$  em algum momento entra no seu estado de aceitação, **aceite**; se  $M$  em algum momento entra em seu estado de rejeição, **rejeite**.”

## Problema da Parada

Não é possível construir uma MT que decida  $A_{MT}$ .

# O Problema da Parada

## Máquina de Turing Universal

É uma MT capaz de simular qualquer outra MT.

A MT  $U$  apresentada anteriormente é uma MT Universal.

# O Problema da Parada

## Máquina de Turing Universal

É uma MT capaz de simular qualquer outra MT.

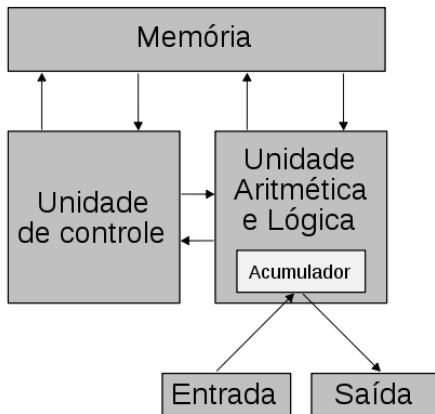
A MT  $U$  apresentada anteriormente é uma MT Universal.

## Contribuição importante

A MT Universal estimulou o desenvolvimento de computadores com programas armazenado.

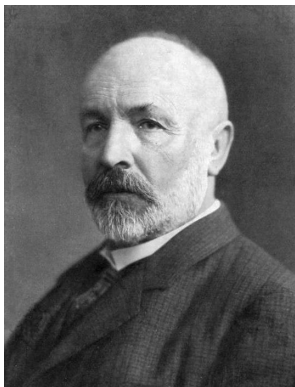


# O Problema da Parada



**Figura:** Arquitetura de von Neumann (1945).

# Método da diagonalização



## Contribuição

Criou o método da diagonalização em 1873.

## Quem?

**George Cantor (1845-1918)**  
Matemático russo.

# Método da diagonalização

## Problema

Se temos dois conjuntos infinitos, como podemos dizer se um conjunto é maior que o outro (ou se eles têm o mesmo tamanho)?

# Método da diagonalização

## Problema

Se temos dois conjuntos infinitos, como podemos dizer se um conjunto é maior que o outro (ou se eles têm o mesmo tamanho)?

## Conjuntos finitos

Podemos utilizar o método da contagem.

# Método da diagonalização

## Problema

Se temos dois conjuntos infinitos, como podemos dizer se um conjunto é maior que o outro (ou se eles têm o mesmo tamanho)?

## Conjuntos finitos

Podemos utilizar o método da contagem.

## Proposta de Cantor

Dois conjuntos finitos têm o mesmo tamanho se os elementos de um deles puder ser emparelhados com os elementos do outro. Basta estendermos essa ideia para os conjuntos infinitos!



# Decidibilidade

Esdras Lins Bispo Jr.  
bispojr@ufg.br

Teoria da Computação  
Bacharelado em Ciência da Computação

08 de junho de 2017