# Definição de Algoritmo

Esdras Lins Bispo Jr.
bispojr@ufg.br

Teoria da Computação
Bacharelado em Ciência da Computação

**16 de abril de 2018**

# Plano de Aula

# Sumário

1. Revisão
   - Máquina de Turing
2. Definição de algoritmo

# Problema

**Problema 3.16 (b)**

Mostre que a coleção de linguagens Turing-reconhecíveis é fechada sob a operação de concatenação.

# Problema

**Prova:** Sejam duas linguagens Turing-reconhecíveis (TR) quaisquer $A$ e $B$. Sejam $M_A$ e $M_B$ as duas máquinas de Turing (MT) que reconhecem $A$ e $B$, respectivamente (pois se uma linguagem é Turing-reconhecível, então uma MT a reconhece). Iremos construir uma MT não-determinística (MTN) $M_{aux}$, a partir de $M_A$ e $M_B$, que reconhece $A \circ B$. A descrição de $M_{aux}$ é dada a seguir:
$M_{aux}$ = "Sobre a entrada $\omega$, faça:

1. Corte, não deterministicamente, $\omega$ em duas cadeias (i.e. $\omega = \omega_1 \circ \omega_2$).
2. Rode $M_A$ sobre $\omega_1$. Se $M_A$ rejeita, *rejeite*.
3. Rode $M_B$ sobre $\omega_2$. Se $M_B$ aceita, *aceite*.
4. *Rejeite*".

Como é possível construir $M_{aux}$, então $A \circ B$ é TR (pois toda MTN tem uma MT equivalente). Logo, a classe de linguagens Turing-reconhecíveis é fechada sob a operação de concatenação ■

# Definição de algoritmo

## Contribuição

Apresentou uma noção do que seria um algoritmo no Congresso Internacional de Matemáticos em Paris, no ano de 1900.

## Quem?

**David Hilbert (1862-1943)**
Matemático alemão.

# Sumário

# Polinômio

## Definições

Um **polinômio** é uma soma de termos. Um **termo** é um produto de variáveis e uma constante chamada de **coeficiente**.

# Polinômio

## Definições

Um **polinômio** é uma soma de termos. Um **termo** é um produto de variáveis e uma constante chamada de **coeficiente**.

## Exemplo: Termo

$6 \cdot x \cdot x \cdot y \cdot z \cdot z \cdot z = 6x^2yz^3$

# Polinômio

## Definições

Um **polinômio** é uma soma de termos. Um **termo** é um produto de variáveis e uma constante chamada de **coeficiente**.

## Exemplo: Termo

$6 \cdot x \cdot x \cdot y \cdot z \cdot z \cdot z = 6x^2yz^3$

## Exemplo: Polinômio

$6x^2yz^3 + 3xy^2 - 10$

UFG
Regional Jataí

# Polinômio

## Definições

Uma **raiz** de um polinômio é uma atribuição de valores às suas variáveis de modo que o valor do mesmo seja 0. Chamamos de **raiz inteira** aquela em todos os valores atribuídos são valores inteiros.

# Polinômio

## Definições

Uma **raiz** de um polinômio é uma atribuição de valores às suas variáveis de modo que o valor do mesmo seja 0. Chamamos de **raiz inteira** aquela em todos os valores atribuídos são valores inteiros.

## Exemplo: Raiz

O polinômio $6x^3yz^2 + 3xy^2 - x^3 - 10$ tem uma raiz em $x = 5, y = 3$ e $z = 0$.

# Polinômio

## Definições

Uma **raiz** de um polinômio é uma atribuição de valores às suas variáveis de modo que o valor do mesmo seja 0. Chamamos de **raiz inteira** aquela em todos os valores atribuídos são valores inteiros.

## Exemplo: Raiz

O polinômio $6x^3yz^2 + 3xy^2 - x^3 - 10$ tem uma raiz em $x = 5, y = 3$ e $z = 0$.

## Exemplo: Raiz Inteira

A raiz do exemplo acima é uma raiz inteira.

# Polinômio

## Problema apresentado por Hilbert

É possível conceber um algoritmo que teste se um polinômio tem uma raiz inteira ou não?

# Polinômio

## Problema apresentado por Hilbert

É possível conceber um algoritmo que teste se um polinômio tem uma raiz inteira ou não?

## Expressão utilizada por Hilbert

"Um processo com o qual ela possa ser determinada por um número finito de operações".

# Polinômio

## Problema apresentado por Hilbert

É possível conceber um algoritmo que teste se um polinômio tem uma raiz inteira ou não?

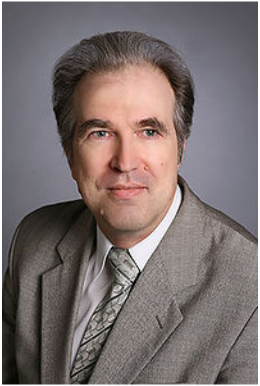
## Expressão utilizada por Hilbert

"Um processo com o qual ela possa ser determinada por um número finito de operações".

## Curioso

Não existe algoritmo que execute esta tarefa.

# Definição de algoritmo



## Contribuição

Mostrou, em 1970, que não existe algoritmo para se testar se um polinômio tem raízes inteiras.

## Quem?

**Yuri Matijasevich (1947-)**
Cientista da computação e matemático russo.

# Definição de Algoritmo

| *Noção intuitiva de algoritmos* | é igual a | *algoritmos de máquina de Turing* |
|---|---|---|

**FIGURA 3.22**
A Tese de Church–Turing

UFG
Regional Jataí

# Definição de Algoritmo

| | | |
|---|---|---|
| *Noção intuitiva de algoritmos* | é igual a | *algoritmos de máquina de Turing* |

**FIGURA 3.22**
A Tese de Church–Turing

## Conclusão

Existem problemas que são algoritmicamente insolúveis.

UFG
Regional Jataí

# Definição de Algoritmo

**Contexto**

$D = \{p \mid p \text{ é um polinômio com uma raiz inteira}\}$

# Definição de Algoritmo

## Contexto

$D = \{p \mid p \text{ é um polinômio com uma raiz inteira}\}$

## Problema

O conjunto $D$ é decidível?

# Definição de Algoritmo

**Contexto**

$D = \{p \mid p \text{ é um polinômio com uma raiz inteira}\}$

**Problema**

O conjunto $D$ é decidível?

**Resposta**

Não é decidível. Mas é Turing-reconhecível.

# Definição de Algoritmo

## Problema análogo

$D_1 = \{p \mid p$ é um polinômio sobre $x$ com uma raiz inteira$\}$

# Definição de Algoritmo

## Problema análogo

$D_1 = \{p \mid p$ é um polinômio sobre $x$ com uma raiz inteira$\}$

## MT $M_1$ que reconhece $D_1$

$M_1$ = "A entrada é um polinômio $p$ sobre a variável $x$.

1. Calcule o valor de $p$ com $x$ substituída sucessivamente pelos valores $0, 1, -1, 2, -2, 3, -3, \ldots$
   Se em algum ponto o valor do polinômio resulta em 0, *aceite*.

# Definição de Algoritmo

## Problema análogo

$D_1 = \{p \mid p$ é um polinômio sobre $x$ com uma raiz inteira$\}$

## MT $M_1$ que reconhece $D_1$

$M_1 =$ "A entrada é um polinômio $p$ sobre a variável $x$.

1. Calcule o valor de $p$ com $x$ substituída sucessivamente pelos valores $0, 1, -1, 2, -2, 3, -3, \ldots$
   Se em algum ponto o valor do polinômio resulta em 0, *aceite*.

## Considerações

$M_1$ reconhece $D_1$, mas não a decide.

# Definição de Algoritmo

**Resultado obtido por Matijasevich**

É possível construir um decisor para $D_1$. Mas não para $D$.

# Definição de Algoritmo

## Resultado obtido por Matijasevich

É possível construir um decisor para $D_1$. Mas não para $D$.

## Justificativa

É possível obter um limitante para polinômios de uma única variável. Porém, Matijasevich provou ser impossível calcular tais limitantes para polinômios multivariáveis.

# Definição de Algoritmo

## Resultado obtido por Matijasevich

É possível construir um decisor para $D_1$. Mas não para $D$.

## Justificativa

É possível obter um limitante para polinômios de uma única variável. Porém, Matijasevich provou ser impossível calcular tais limitantes para polinômios multivariáveis.

## Limitante para polinômios de uma única variável

$$\pm k \frac{c_{max}}{c_1}$$

em que

- $k$ é o número de termos do polinômio,
- $c_{max}$ é o coeficiente com maior valor absoluto, e
- $c_1$ é o coeficiente do termo de mais alta ordem.

UFG
Regional Jataí

# Terminologia para descrever MTs

## Níveis de descrição

- **Descrição formal**: esmiúça todos os elementos da 7-upla, conforme definição;

# Terminologia para descrever MTs

## Níveis de descrição

- **Descrição de implementação**: descreve a forma pela qual a MT move a sua cabeça e a forma como ela armazena os dados na fita;

# Terminologia para descrever MTs

## Níveis de descrição

- **Descrição de alto nível**: neste nível não precisamos mencionar como a máquina administra a sua fita ou sua cabeça de leitura-escrita.

# Terminologia para descrever MTs

## Níveis de descrição

- **Descrição formal**: esmiúça todos os elementos da 7-upla, conforme definição;
- **Descrição de implementação**: descreve a forma pela qual a MT move a sua cabeça e a forma como ela armazena os dados na fita;
- **Descrição de alto nível**: neste nível não precisamos mencionar como a máquina administra a sua fita ou sua cabeça de leitura-escrita.

# Exemplo

Seja $A$ a linguagem consistindo em todas as cadeias representando grafos não-direcionados que são conexos. Logo:

$$A = \{\langle G \rangle | G \text{ é um grafo não-direcionado conexo}\}$$

# Exemplo

Seja $A$ a linguagem consistindo em todas as cadeias representando grafos não-direcionados que são conexos. Logo:

$$A = \{\langle G \rangle | G \text{ é um grafo não-direcionado conexo}\}$$

## Descrição de alto nível

$M =$ "Sobre a entrada $\langle G \rangle$, a codificacão de um grafo $G$:

1. Selecione o primeiro nó de $G$ e marque-o.
2. Repita o seguinte estágio até que nenhum novo nó seja marcado:
    1. Para cada nó em $G$, marque-o se ele está ligado por uma aresta a um nó que já está marcado.
3. Faça uma varredura em todos os nós de $G$ para determinar se eles estão todos marcados. Se eles estão, *aceite*; caso contrário, *rejeite*".

# Exemplo

**Pergunta**

Como seria a descrição de $M$ no nível de implementação?

# Definição de Algoritmo

Esdras Lins Bispo Jr.
bispojr@ufg.br

Teoria da Computação
Bacharelado em Ciência da Computação

**16 de abril de 2018**

UFG
Regional Jataí