Definição de Algoritmo e Decidibilidade

Esdras Lins Bispo Jr.
bispojr@ufg.br

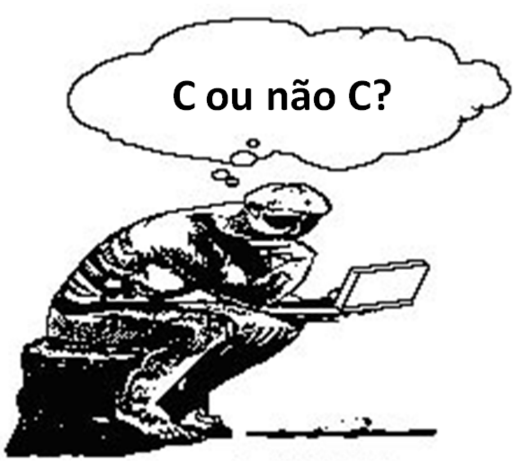
Teoria da Computação
Bacharelado em Ciência da Computação

24 de maio de 2016

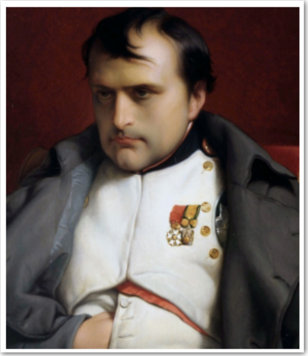
Plano de Aula

Sumário

Pensamento



Pensamento



Frase

Nada é mais difícil e, portanto, tão precioso, do que ser capaz de decidir.

Quem?

Napoleão Bonaparte (1769-1821)

Imperador francês.

Bônus (0,5 pt)

Desafio

- **Problema 4.7:**
  Seja $T = \{(i, j, k) | i, j, k \in \mathbb{N}\}$. Mostre que T é contável.
- Candidaturas até o fim da aula (24 de maio, 11h00);
- Apresentação e resposta por escrito $\rightarrow$
  Segunda (07 de junho, 11h30);
- 20 minutos de apresentação.

Candidato

???

Sumário

UFG
Regional Jataí

MT Multifita

Teorema

Toda máquina de Turing multifita tem uma máquina de Turing de uma única fita que lhe é equivalente.

Corolário

Uma linguagem é Turing-reconhecível se e somente se alguma máquina de Turing multifita a reconhece.

MT Multifita

PROVA Uma linguagem Turing-reconhecível é reconhecida por uma máquina de Turing comum (com uma única fita), o que é um caso especial de uma máquina de Turing multifita. Isso prova uma direção desse corolário. A outra direção segue do Teorema 3.13.

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.

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.

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?

Expressão utilizada por Hilbert

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

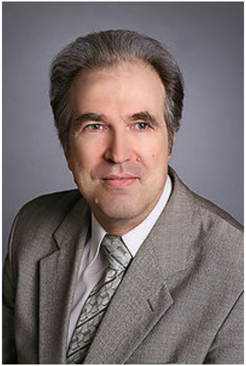
Curioso

Não existe algoritmo que execute esta tarefa.

Sumário

UFG
Regional Jataí

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

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.

Definição de Algoritmo

Contexto

$D = \{p \mid p$ 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

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

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*.

UFG
Regional Jataí

Definição de Algoritmo

Problema análogo

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

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.

UFG
Regional Jataí

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*".

UFG
Regional Jataí

Exemplo

Pergunta

Como seria a descrição de M no nível de implementação?

Sumário

Introdução

Propósitos da Teoria da Computação

- Conhecer o poder dos algoritmos;
- Explorar os limites da solubilidade algorítmica;
- Identificar algoritmos insolúveis.

Introdução

Propósitos da Teoria da Computação

- Conhecer o poder dos algoritmos;
- Explorar os limites da solubilidade algorítmica;
- Identificar algoritmos insolúveis.

Por que devemos estudar insolubilidade?

- Relaxamento dos requisitos;
- Conhecimento das limitações dos modelos computacionais.

UFG Regional Jataí

Linguagens Decidíveis

Exemplos de Linguagens Decidíveis

São úteis porque

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

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 ω, identificar se MC aceita ω.

UFG
Regional Jataí

Problema da aceitação para AFDs

Problema da aceitação para AFDs

Dado um AFD B e uma cadeia de entrada ω, identificar se B aceita ω.

Problema da aceitação para AFDs

Problema da aceitação para AFDs

Dado um AFD B e uma cadeia de entrada ω, identificar se B aceita ω.

Problema

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

Problema da aceitação para AFDs

Problema da aceitação para AFDs

Dado um AFD B e uma cadeia de entrada ω, identificar se B aceita ω.

Problema

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

Estratégia de Resolução

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

UFG Regional Jataí

Problema da aceitação para AFDs

Teorema 4.1

A_{AFD} é uma linguagem decidível.

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 ω, uma cadeia:

1. Simule B sobre a entrada ω;
2. Se a simulação termina em um estado de aceitação, **aceite**. Senão, **rejeite**."

UFG
Regional Jataí

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, Σ, δ, q_0 e F;
  - M simula B de forma que M **aceita** se B estiver em um estado final, e **rejeita**, caso contrário.

UFG
Regional Jataí

Problema da aceitação para AFNs

Problema da aceitação para AFNs

Dado um AFN B e uma cadeia de entrada ω, identificar se B aceita ω.

Problema da aceitação para AFNs

Problema da aceitação para AFNs

Dado um AFN B e uma cadeia de entrada ω, identificar se B aceita ω.

Problema

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

Problema da aceitação para AFNs

Problema da aceitação para AFNs

Dado um AFN B e uma cadeia de entrada ω, identificar se B aceita ω.

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

UFG
Regional Jataí

Problema da aceitação para AFNs

Teorema 4.2

A_{AFN} é uma linguagem decidível.

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 ω, 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**."

UFG
Regional Jataí

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

UFG
Regional Jataí

Problema da Vacuidade de uma Linguagem

Teorema 4.4

V_{AFD} é uma linguagem decidível.

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**."

Definição de Algoritmo e Decidibilidade

Esdras Lins Bispo Jr.
bispojr@ufg.br

Teoria da Computação
Bacharelado em Ciência da Computação

24 de maio de 2016

UFG
Regional Jataí