

# Algoritmo

- amontoado de instruções simples  
p/ processar informação  
a fim de obter resultado

- matemático? culinária? engenharia

- sequência de passos que resolve um problema

---

$$2x - 1 = 7 \rightsquigarrow 2x = 8 \rightsquigarrow x = 4$$

séc VIII: Al-Khwarizmi (Algorithmi) "Al-jabr"

∴ Bastante coisa: Pascal, Leibniz, Babbage  
∴  
séc XIX/XX ↪ interesse em formalizar a matemática

— Frege: Begriffsschrift

— Emil du Bois-Reymond: "ignoramus et ignorabimus"

— Definição de conjunto

↳ usando "propriedades"

↳ Russell: paradoxo, contradição:  $\varphi(x): x \notin x$   
define R

- Hilbert

↳ Congresso dos Matemáticos de 1900 (Paris)

↳ Os 23 problemas de Hilbert

1900-1930 ↳ O Programa de Hilbert

↳ formalizar toda a matemática em um único sistema formal

"claro e preciso"

↳ "Problema da Decisão": existe um método (algoritmo) para decidir, dada uma fórmula, se ela pode ser provada?

Entscheidungsproblem

- Königsberg 1930 : palestra de aposentadoria de Hilbert

↳ no mesmo congresso, Gödel anuncia seu Teorema de Incompletude

↳ no "processo", Gödel define a noção de Função Recursiva Geral

→ Gödel publica em 1931

- Church :  $\lambda$ -cálculo, noção de "função efetivamente calculável"

↳ prova (1936) que o Problema da decisão tem resposta NÃO : primeiro, tem que haver uma formalização de "algoritmo" (p/ Church, "efetivamente calculável")

Informal

"MÉTODO"

"PASSO A PASSO"

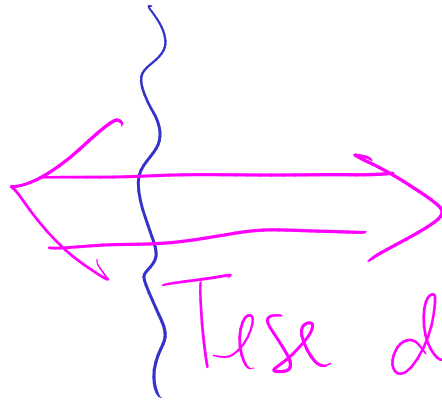
Formal

Gödel

Turing

Church

Tese de Church-Turing



- Turing, 1936: mais uma noção formal de "algoritmo"

↳ usando abordagem mais "pé no chão": máquina

↳ Equivalente a Church & Gödel!

↳ Extremamente influente no desenvolvimento dos computadores

- Turing provou que há problemas que suas máquinas não resolvem. O primeiro é o Problema da Parada

# Teoria da Computabilidade

Decidíveis

Indecidíveis

Estutura riquíssima

→ quão rápido?

Teoria da Complexidade

→ SAT (Teorema de Cook-Levin)

# Máquinas de Turing

Modelo ROBUSTO: podemos mudar vários detalhes sem afetar a noção de computabilidade obtida

Elementos: 1) uma fita dividida em células, infinita (apenas) para a direita.

- Cada célula contém no máximo um símbolo de um alfabeto fixo *definido com a máquina*
- a primeira célula contém um símbolo especial ▷



- apenas uma quantidade finita de células  
está preenchida a qualquer momento

2) Um "cabeçote" de leitura/escrita

- sempre sobre exatamente uma célula

- pode ler e escrever (apagar) na fita

- pode se mover para a direita (sempre)

ou para a esquerda (exceto quando está sobre  $\triangleright$ )

- o símbolo  $\triangleright$  não pode ser apagado nem escrito  
em outras células

3) Um conjunto finito de estados ("modos")

4) A "transição" (funcionamento dinâmico): dado o estado atual da máquina e o símbolo lido pelo cabeçote, a máquina pode:

a) mudar (ou não) de estado ; e

b) escrever um novo símbolo ou mover o cabeçote

← apagando o anterior

↑  
exclusivo (para nós)

Exercício informal: Volar uma máquina que começa numa fita com alguma quantidade de "1"s escrita no começo da fita e dobra essa quantidade de "1"s, usando apenas o alfabeto

$\{1, 2, 3\}$

▷ 1 1 1

⋮

▷ 1 1 1 1 1 1