



**PCS3616**

# Programação de Sistemas

(Sistemas de Programação)

**Visão Geral**

Escola Politécnica da Universidade de São Paulo

# Objetivos

- Familiarização com o funcionamento global de software de sistema computacional
- Desenvolver experimentos em computador
  - Ambiente (Expressões Regulares, scripts)
  - Máquinas/Algoritmos (Máquina de Turing)
  - Software de Sistema (Montadores, Editores de Ligação. Relocadores)
  - Noções de Compiladores (Front-end)
- Ferramenta para automatização de geração de código, otimização em Compiladores (Back-end).

# Organização

- Material no Moodle do Stoa
  - <https://disciplinas.stoa.usp.br/course/view.php?id=31356>
- Professor
  - Ricardo Luis de Azevedo da Rocha
- Monitor
  - ???

# Bibliografia

## ▪ Básica

- SAVAGE, J. “Models of Computation: Exploring the Power of Computing”,
- BRYANT R. E. and O’HALLARON, D. R. “Computer Systems: A Programmer’s Perspective”, 2010
- DONOVAN, J. “Systems Programming”, 1972
- RICARTE, I. “Introdução à Compilação”, 2008
- JOSÉ NETO, J. “Introdução à Compilação”, 1987
- LLVM – [llvm.org](http://llvm.org); [www.aosabook.org/en/llvm.html](http://www.aosabook.org/en/llvm.html)

## ▪ Complementar

- LUCCHESI, C. L. “Aspectos Teóricos da Computação”, 1979
- SIPSER, Michael “Introdução à teoria da computação”, Thomson, 2007.
- BECK, L. L. “An introduction to Systems Programming”, Addison Wesley, 1997
- LOPES, B. C. and AULER, R. “Getting started with LLVM Core Libraries”, 2014
- PANDEY, M. and SARDA, S. “LLVM Cookbook”, 2015

# Computação

- Entscheidungsproblem:
  - Início do século (1900), 10º problema de Hilbert
  - Dada uma questão qualquer, como decidir em um número finito de passos se há ou não solução para esta questão.

Comentário: Está embutida a idéia de algoritmo

- O sonho !
  - Solução automática de problemas em um dispositivo qualquer

# Condições Necessárias

## Filosofia

Conceitos: \_\_\_\_\_

- Máquinas algorítmicas
- Inteligência Inteligentes
- Consciência para ajuda

## Matemática

Definição de: \_\_\_\_\_

- Algoritmo, decidibilidade
- Avaliação inteligente
- Avaliação consciente

## Tecnologia

Construção de: \_\_\_\_\_

- Máquinas
- Máquinas
- Máquinas

- Ainda não se possui um conceito completamente aceito para consciência, portanto ainda há um longo caminho teórico a percorrer.

# Desenvolvimento Teórico

- Filosofia
  - Conceito de máquina
- Matemática
  - Definição formal de Algoritmo
    - Procedimento de refutação (Herbrand, 1926)
    - Exposição do conceito de Indecibilidade (Gödel, 1931), nº de Gödel para codificação de símbolos em números
    - Máquina de Turing (Turing, 1936),  $\lambda$ -calculus (Church, 1936), funções  $\mu$ -recursivas (Kleene, 1936), Tese de Church - algoritmo é qualquer rotina fixa executável (cálculo) em uma máquina de Turing
- Tecnologia
  - Possibilidade Teórica de construir máquinas

# Conceito de Computação

$Z = F(x)$ ; Se  $F$  puder ser dividida em sub-funções

$$Z = f_1(f_2(g(x))); F(x) = f_1(f_2(g(x)))$$

. Continuando  
. o  
. processo

$$Z = f_1(f_2(\dots(f_{n-1}(f_n(x))\dots))$$

$\therefore F(x) = f_1(f_2(\dots(f_{n-1}(f_n(x))\dots))$ ; caso  $f_i$  ( $i=1..n$ ), seja instrução executável em alguma máquina ...



# Funções Efetivamente Calculáveis

- O problema proposto por Hilbert passa pela definição (formal ou quase formal) de função efetivamente calculável (à época procedimento efetivamente calculável, R. Gandy *The confluence of ideas in 1936*).
- Uma função efetivamente calculável é aquela que o ser humano pode calcular através de um procedimento automático (mecânico segundo conceito da época).
- O grande problema foi justamente estabelecer uma definição formal para função efetivamente calculável. A máquina de Turing foi proposta como resposta ao problema.

# Funções Efetivamente Computáveis

- Dada uma máquina de Turing  $\underline{M}$ , se uma função qualquer  $\phi$  puder ser executada nesta máquina, então esta função é dita efetivamente computável.
  - Logo esta função também pode ser executada através de funções  $\mu$ -recursivas ou  $\lambda$ -calculus (e também gramáticas)
- Logo o conceito de computação pode ser definido através da Máquina de Turing.

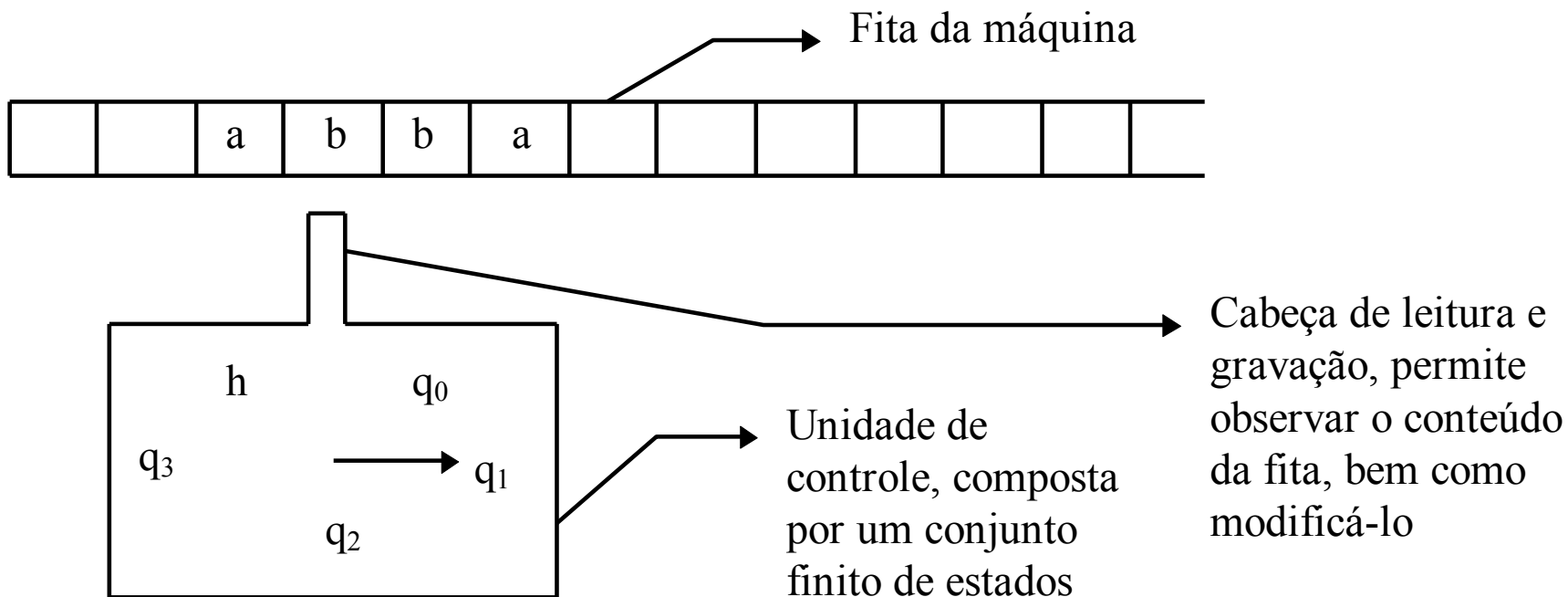
# Funções Computáveis

- Uma função  $f(x_1, \dots, x_n)$  é dita computável se existe uma máquina de Turing  $Z$  tal que produza como resultado uma expressão  $\Psi(x_1, \dots, x_n)$  em sua fita, e que:

$$f(x_1, \dots, x_n) = \Psi(x_1, \dots, x_n)$$

- Esta é a formulação proposta por Turing para o problema das funções efetivamente calculáveis.
- A caracterização de Kleene para o mesmo problema, chamada de funções  $\mu$ -recursivas é baseada em números de Gödel e em aplicações de funções.

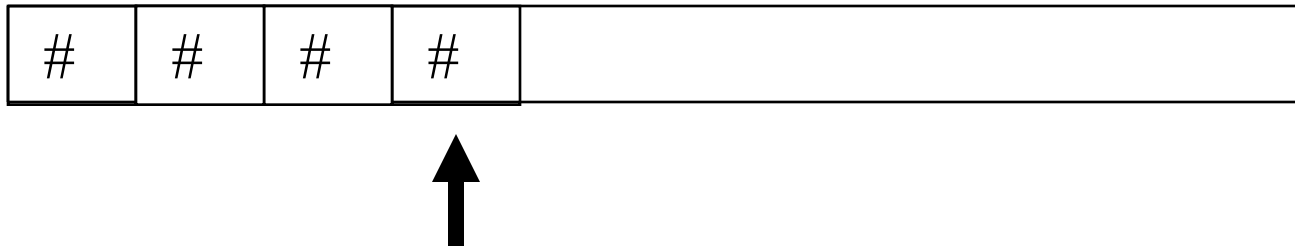
# Máquina de Turing



# Ações da Máquina de Turing

- As ações possíveis (instruções) são:
  - R: Mover a cabeça de leitura/gravação para a esquerda
  - L: Mover a cabeça de leitura/gravação para a direita
  - $\sigma$ : Escreve o símbolo “ $\sigma$ ” na fita, na posição atual da cabeça
- A parada da máquina se dá quando esta entra no estado  $\underline{h}$ , “halt”.

# Exemplo



$M = (K, \Sigma, \delta, s); K = \{q, q'\}; \Sigma = \{a, \#\}; s = q$

$\delta =$	$q$	$\sigma$	$\delta(q, \sigma)$
	$q$	$a$	$(q', \#)$
	$q$	$\#$	$(h, \#)$
	$q'$	$a$	$(q, a)$
	$q'$	$\#$	$(q, R)$



# Máquinas baseadas em Estado

## ☐ Autômatos

- ☐ Autômatos finitos e autômatos de pilha

- ☐ Autômatos adaptativos

- ☐ Equivalência entre a máquina de Turing e o autômato adaptativo, quanto ao poder computacional

## ☐ Máquina de Turing

## ☐ Máquina de Post

# Tese de Church

- Em 1936 Kleene demonstra a equivalência entre lambda-calculi, funções recursivas e máquinas de Turing.
- De posse dos resultados, Church convence-se de que:
  - **Todos os modelos conceituais de computação são equivalentes.**
  - **Aceita-se então que Máquinas de Turing são versões formais de algoritmos.**



# Conclusão

- Desde a proposta de Hilbert em 1900 muito esforço teórico foi dispendido para que se pudesse atingir os resultados atuais, com computadores e alta tecnologia.
- No esforço vários modelos de computação distintos foram desenvolvidos independentemente pelos pesquisadores.
- Demonstra-se de equivalência de poder computacional entre os diversos modelos de computação.

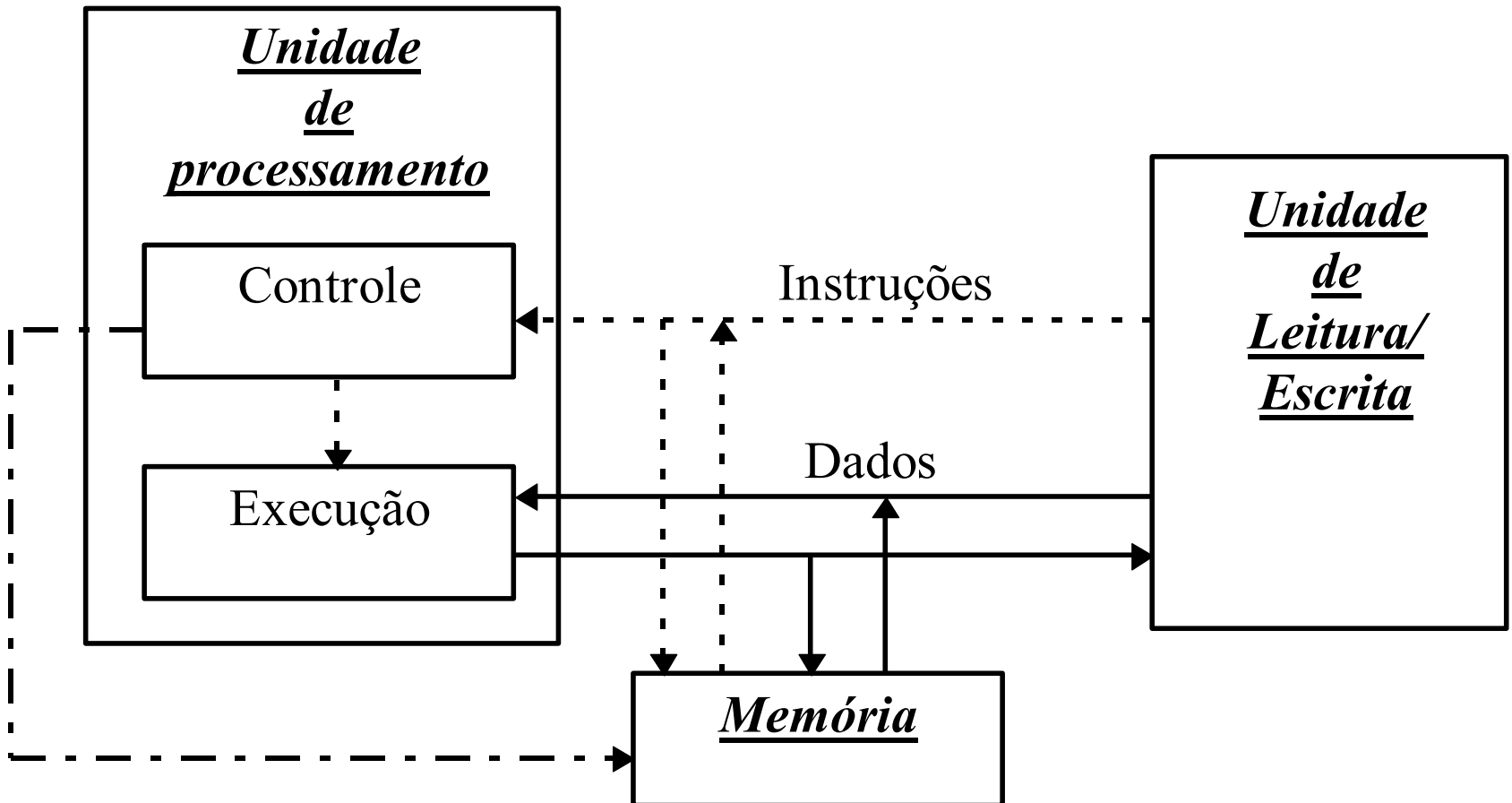
# Conclusão - final

- Modelos de computação
  - Máquinas de Turing
  - Lambda-calculi
  - Gramáticas
  - Funções recursivas
- A Tese de Church propõe que qualquer modelo computacional seja equivalente aos demais.

# De Máquinas de Turing a Computadores Eletrônicos

- A definição de máquina de Turing universal pressupõe que tanto o controle (programa, algoritmo) quanto as informações a serem manipuladas sejam lidas pela máquina (estejam presentes na fita).
- Esta é a definição formal de von Neuman (1945) para o modelo de arquitetura de computadores até hoje utilizado. Baseado na idéia de McCulloch e Pitts de redes neurais (1943).

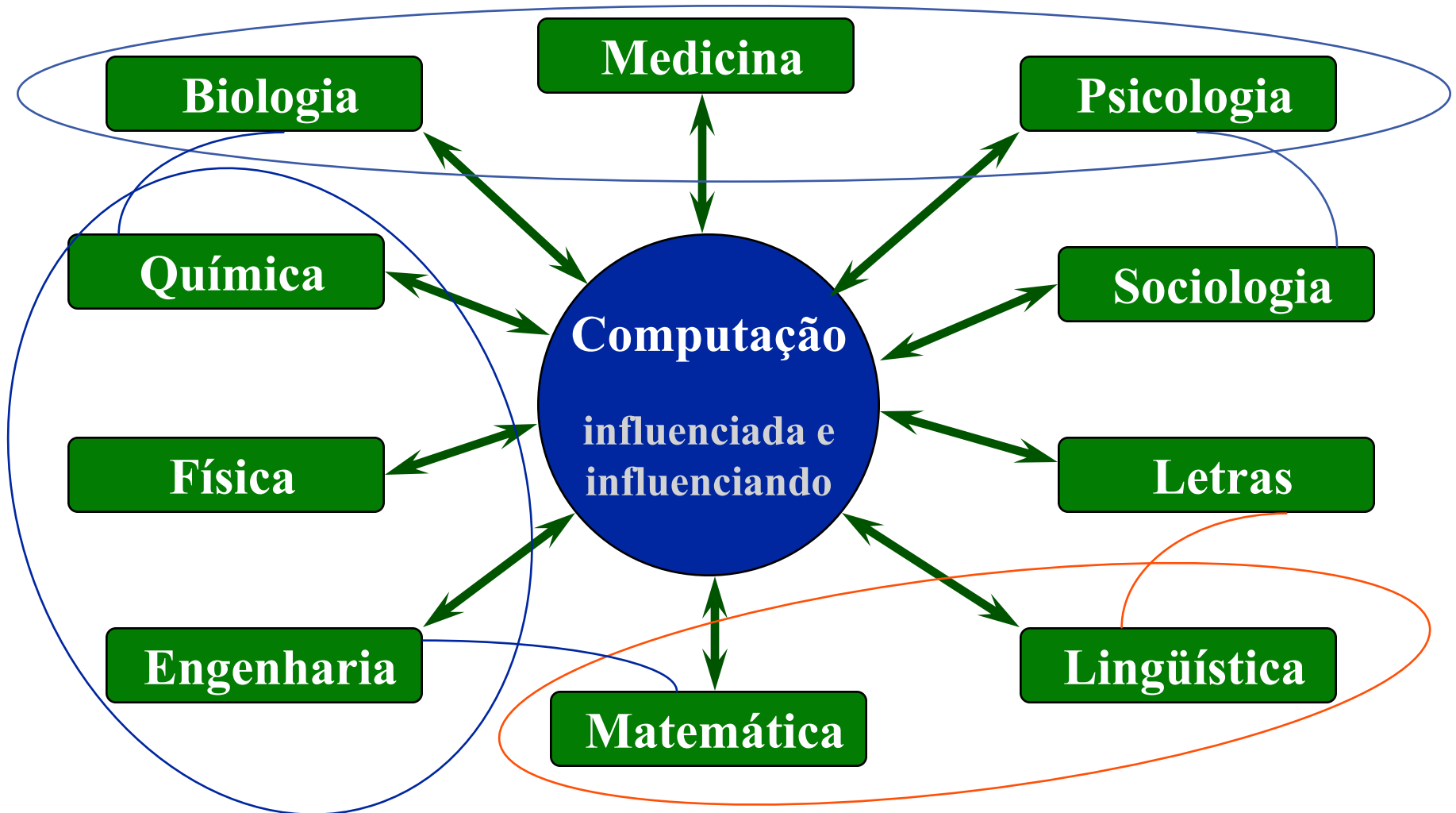
# Arquitetura de von Neuman



# O Computador

- O Computador Eletrônico nada mais é do que um dispositivo que realiza tarefas (rapidamente), desde que as tarefas possam ser divididas em pequenas instruções, e essas instruções sejam dispostas e ordenadas (por um elemento humano), de forma a poderem ser processadas nele.
- As inovações nos computadores são em sua tecnologia de materiais e não em arquitetura.

# Conexão com outras áreas



# Aplicações

- **Computação Gráfica/Realidade Virtual:**
  - Medicina, Simulação, Treinamento, Educação
- **Telecomunicações/Redes/Internet:**
  - Geoprocessamento, Comunicação, Meteorologia
- **Banco de Dados/Hipermídia**
  - Educação, Administração, Direito, Pesquisa, Medicina
- **Tecnologias Inteligentes/I.A.**
  - Autocontrole em dispositivos, Apoio à decisão

# Influência Sofrida

- **Biologia, Sociologia**
  - Robótica, Visão, Trabalho em equipe
- **Psicologia**
  - Inteligência, Consciência
- **Física/Química**
  - Limites Estruturais
- **Matemática/Lingüística/Letras**
  - Limites Teóricos (Gödel), Interfaces, Linguagens



# Conclusão

- Partindo-se de um conceito filosófico, passando por definições matemáticas, chega-se até a tecnologia dos computadores.
- O percurso evolutivo é sempre o mesmo em qualquer área: Princípio Filosófico, Formalização, Aplicação.
- As redes neurais de McCulloch e Pitts são na realidade mais simples que a máquina de Turing, são baseadas em autômatos finitos.
- Só foram resolvidos problemas algorítmicos.