



UNIVERSIDADE DE SÃO PAULO  
INSTITUTO DE CIÊNCIAS MATEMÁTICAS E DE COMPUTAÇÃO  
Departamento de Ciências de Computação

# SCC-0205

## Teoria da Computação e Linguagens Formais

João Luís Garcia Rosa<sup>1</sup>

<sup>1</sup>Departamento de Ciências de Computação  
Instituto de Ciências Matemáticas e de Computação  
Universidade de São Paulo - São Carlos  
<http://www.icmc.usp.br/~joaoluis/>  
[joaoluis@icmc.usp.br](mailto:joaoluis@icmc.usp.br)

2012



# Sumário

- 1 Teoria da Computação e Linguagens Formais
  - A disciplina SCC-0205
  - Objetivos e Programa
  - Avaliação

# Sumário

- 1 Teoria da Computação e Linguagens Formais
  - A disciplina SCC-0205
  - Objetivos e Programa
  - Avaliação

# A disciplina SCC-0205

- A disciplina é composta de três partes centrais da Teoria da Computação que têm o objetivo de tentar responder quais são as capacidades e as limitações dos computadores:
  - 1 Teoria das Linguagens Formais e dos Autômatos,
  - 2 Teoria da Computabilidade e
  - 3 Teoria da Complexidade.

# A disciplina

- A primeira parte (**Teoria das Linguagens Formais e dos Autômatos**) trata das definições e propriedades de modelos matemáticos de computação que têm um papel fundamental em várias áreas da Computação como o processamento de textos, compiladores, definição de linguagens de programação, dentre outras.
- Além desse lado prático, do ponto de vista teórico, para se definir o que é ou não computável é necessário utilizar um modelo matemático que represente o que se entende por computação.

# A disciplina

- A segunda parte do curso (**Teoria da Computabilidade**) é centralizada na Tese de Church-Turing e nas evidências dela.
- Church usou um sistema chamado cálculo- $\lambda$  para definir algoritmo e Turing fez o mesmo com o uso da Máquina de Turing (MT).
- As duas definições foram mostradas serem equivalentes e a conexão entre a noção informal de algoritmo (solúvel efetivamente) e a definição precisa por uma MT foi chamada Tese de Church-Turing: se um problema algorítmico não pode ser resolvido por uma máquina de Turing, então não existe nenhuma solução computável para ele.

# A disciplina

- Vários outros modelos de computação (por exemplo, as funções recursivas de Kleene, linguagens formais, RAMs, algoritmos de Markov, linguagens de programação, a máquina de Post) foram propostos e provados terem poder equivalente à máquina de Turing.
- Assim, estudando qualquer um destes modelos, por exemplo um modelo simples como a máquina de Turing, é possível aprender sobre as limitações teóricas de todos os computadores.
- A meta da teoria da computabilidade é a classificação de problemas em solúveis, parcialmente solúveis e insolúveis e se forem problemas de decisão, em problemas decidíveis, parcialmente decidíveis e indecidíveis.

# A disciplina

- Nem todos os problemas algorítmicos, que podem ser resolvidos em princípio, podem ser resolvidos na prática: os recursos computacionais requeridos (tempo ou espaço) podem ser proibitivos.
- Esta observação motiva o estudo da complexidade computacional que será tratada na terceira parte do curso (**Teoria da Complexidade**).
- A meta principal da teoria da complexidade é a classificação de problemas de acordo com a dificuldade computacional.



# Sumário

- 1 Teoria da Computação e Linguagens Formais
  - A disciplina SCC-0205
  - **Objetivos e Programa**
  - Avaliação

# Objetivos e Justificativa

- **Objetivos:**
  - Apresentar ao aluno conceitos fundamentais das disciplinas de teoria da computação, linguagens formais e lógica.
  - Capacitar o aluno a compreender e utilizar estes conceitos.
- **Justificativa:**
  - O estudo destes aspectos fundamentais da ciência da computação deve auxiliar na formação da base teórica necessária às demais disciplinas do curso.

# Programa

## 1 Linguagens Regulares e Autômatos Finitos

### ① Gramáticas e Linguagens

- A Primeira Linguagem
- Gramáticas e Linguagens
- Linguagens Regulares e de Estados Finitos

### ② Autômatos de Estados Finitos

- Autômatos Finitos
- Arcos- $\lambda$
- Autômato Mínimo

### ③ Autômatos Finitos com Saída

- Máquinas de Mealy
- Máquinas de Moore
- Exemplos

# Programa

## 2 Linguagens Livres de Contexto e Autômatos de Pilha

### 1 Linguagens Livres de Contexto

- Linguagens Livres de Contexto
- Lema do Bombeamento para Linguagens Livres de Contexto
- Formas Normais para Gramáticas Livres de Contexto

### 2 Autômatos de Pilha

- A Pilha como Processador de Linguagem
- O Autômato de Pilha
- O Teorema da Equivalência

### 3 Programas, Linguagens e *Parsing*

- Linguagens de Programação
- *Parsing*
- Gramáticas Livres de Contexto e a Língua Natural

# Programa

## 3 Linguagens Sensíveis ao Contexto e Autômatos Limitados Linearmente

- ① Gramáticas e Linguagens Sensíveis ao Contexto
  - Gramáticas e Linguagens Sensíveis ao Contexto
  - O Lema da Cadeia Vazia
  - Prova do Lema da Cadeia Vazia
- ② Máquinas de Turing
  - Máquinas de Turing e a Computabilidade
  - Conjunto de Aceitação de uma Máquina de Turing
- ③ Autômatos Limitados Linearmente
  - Autômatos Limitados Linearmente
  - O Lema do Alfabeto

# Programa

## 4 Linguagens Recursivamente Enumeráveis e Máquinas de Turing

### ① Gramáticas Irrestritas

- Gramáticas Irrestritas
- Das Gramáticas para as Máquinas de Turing
- Das Máquinas de Turing para as Gramáticas

### ② A Máquina de Turing Universal

- A Máquina de Turing e Funções Numéricas
- A Tese de Church-Turing
- A Máquina Universal

# Programa

## 5 Computabilidade e Complexidade

### 1 Indecidibilidade

- Máquinas de Turing Não Determinísticas
- Uma Linguagem que não é Recursivamente Enumerável
- O Problema da Parada e a Indecidibilidade

### 2 Teoria de Complexidade

- Complexidade de Tempo
- Complexidade de Espaço

### 3 Tratabilidade e Problemas $\mathcal{NP}$ -Completo

- Tratabilidade
- A Classe  $\mathcal{NP}$
- Outras Classes de Problemas

### • Aulas:

- Terças: 8h10-9h50
- Quintas: 10h10-11h50
- Sala 5-004

# Sumário

- 1 Teoria da Computação e Linguagens Formais
  - A disciplina SCC-0205
  - Objetivos e Programa
  - Avaliação



# Avaliação

- 3 provas:
  - $P_1 = 30/8$
  - $P_2 = 16/10$
  - $P_3 = 29/11$
- 2 trabalhos em grupo, com implementação:
  - Submissão do Trabalho  $T_1$ : 27/9.
  - Submissão do Trabalho  $T_2$ : 22/11.
- Trabalho Extra (Alan Turing) - opcional  $T_3$ : 29/11
  - acréscimo de 0,5 na média.

# Avaliação

- MP = Média Ponderada das Provas:
  - $MP = P_1 * 0,4 + P_2 * 0,3 + P_3 * 0,3$
- MT = Média Aritmética dos Trabalhos
- MF = Média Final:
  - Se  $MP \geq 5,0$  e  $MT \geq 5,0$  então  $MF = (7*MP + 3*MT)/10$
  - Senão MF = menor valor entre MP e MT

# Avaliação: Recuperação

## ● Norma de Recuperação

- 1 prova de recuperação  $P_R$
- Realização: Até a primeira semana de aulas do semestre posterior.
- Critério de Aprovação:
  - Média =  $MF + (P_R/2, 5)$ , se  $P_R \geq 7,5$ ; ou
  - Média =  $Max\{MF, P_R\}$ , se  $P_R < 5,0$ ; ou
  - Média =  $5,0$ , se  $5,0 \leq P_R < 7,5$ .

# Integridade Acadêmica

- A “cola” ou plágio em provas, exercícios ou atividades práticas implicará na atribuição de nota zero para todos os envolvidos. Dependendo da gravidade do incidente, o caso será levado ao conhecimento da Coordenação e do Conselho do Departamento, para as providências cabíveis. Na dúvida do que é considerado cópia ou plágio, o aluno deve consultar o professor antes de entregar um trabalho.

Ano Alan Turing: ★1912 † 1954



**2012 THE ALAN TURING YEAR**

A Centenary Celebration of the Life and Work of Alan Turing

# Bibliografia Básica I






[1] Rosa, J. L. G.



*Linguagens Formais e Autômatos.*

Editora LTC, 2010.

# Bibliografia Complementar I

-  [1] Hopcroft, J. E., Ullman, J. D.  
*Formal Languages and Their Relation to Automata.*  
Addison-Wesley Publishing Company, 1969.
-  [2] Hopcroft, J. E., Ullman, J. D. e Motwani, R.  
*Introdução à Teoria de Autômatos, Linguagens e Computação.*  
Tradução da segunda edição americana. Editora Campus, 2003.
-  [3] JFLAP Version 6.0.  
Ferramenta para Diagrama de Estados.  
[www.jflap.org](http://www.jflap.org).

# Bibliografia Complementar II

-  [4] Moll, R. N., Arbib, M. A., and Kfoury, A. J.  
*An Introduction to Formal Language Theory.*  
Springer-Verlag, 1988.
-  [5] Sipser, M.  
*Introduction to the Theory of Computation.*  
Second Edition, Thomson, 2006.