

# Aspectos Teóricos da Computação

Prof. Rodrigo Martins

[rodrigo.martins@francomontoro.com.br](mailto:rodrigo.martins@francomontoro.com.br)

# Cronograma da Aula

- Aula Inaugural
- Apresentação da disciplina
- Critérios de Avaliação
- Metodologia de Trabalho

# Ementa

- Gramáticas, máquinas de estados finitos, autômatos, expressões regulares e suas aplicações em linguagens formais, compiladores, síntese de circuitos sequenciais, protocolos de comunicação.

# Objetivos da Disciplina

- Proporcionar uma visão abrangente dos fundamentos da teoria dos processos efetivos (algoritmos e programas), por meio do estudo de vários modelos formais de computação, como base para a especificação e a análise de algoritmos.
- Dentre as aptidões, competências e habilidades esperadas vislumbra-se: projetar e implementar algoritmos básicos por meio de autômatos finitos; formular gramáticas como meio de geração e reconhecimento de linguagens; desenvolver árvores de derivação para reconhecimento de vários tipos de linguagem artificiais ou naturais; relacionar o tipo de gramática ao tipo de linguagem a ser produzido ou reconhecido

# Critérios de Avaliação

T1 – Lista de Exercícios/Trabalhos/Projetos (30%)

P1 - Avaliação Bimestral (P1) (70%)

Média 1º Bimestre:  $T1 * 0.30 + P1 * 0.70$

T2 – Lista de Exercícios/Trabalhos/Projetos (30%)

P2 - Avaliação Bimestral

Média 2º Bimestre:  $T2 * 0.30 + P2 * 0.70$

Média Final –  $(M1ºB + M2ºB) / 2$

# Metodologia de Trabalho

- Aulas expositivas com exercícios.
- Uso do simulador JFLAP.
- Lista de exercícios para serem resolvidas para fixação dos assuntos abordados nas aulas expositivas.
- Uso de metodologias ativas

# Bibliografia Básica

- HOPCROFT, J.E.; MOTWANI, R.; ULLMAN, J.D. Introdução a Teoria de Autômatos, Linguagens e Computação. Campus, 2002.
- PAPADIMITRIOU, C., HARRY, L., Elementos de Teoria da Computação, Bookman, 2000.
- MENEZES, P. B. Linguagens Formais e Autômatos. Editora SagraLuzzato, 2000.

# Bibliografia Complementar

- MENEZES, P. F.B., DIVERIO, T.A., Teoria da Computação, Sagra-Luzzatto, 1999.



# Porque estudar aspectos teóricos da computação?

- Os aspectos teóricos da computação são fundamentais no campo da ciência da computação por várias razões, atuando como a espinha dorsal que sustenta muitos dos avanços e inovações tecnológicas.
- Os próximos 7 slides apresentam algumas das principais razões para sua importância.

# 1. Base para Entendimento de Algoritmos e Estruturas de Dados

- Os conceitos teóricos fornecem a base para entender como algoritmos e estruturas de dados funcionam e são otimizados.
- Crucial para o desenvolvimento de software eficiente e para a solução de problemas complexos.

## 2. Fundamentos de Linguagens de Programação

- A teoria da computação aborda as linguagens formais e autômatos, que são essenciais para o design e análise de linguagens de programação.
- Entender esses conceitos ajuda a compreender como diferentes linguagens funcionam e são processadas.

### 3. Desenvolvimento e Análise de Compiladores

- A construção de compiladores depende fortemente da teoria da computação.
- Compreender a teoria por trás dos compiladores permite a criação de compiladores mais eficientes e a tradução eficaz de linguagens de programação de alto nível para código de máquina.

## 4. Complexidade Computacional e Teoria da Decisão

- A teoria da computação ajuda a entender a complexidade dos problemas e algoritmos, permitindo aos cientistas da computação classificar problemas como resolvíveis ou insolúveis e determinar a eficiência dos algoritmos.

## 5. Pesquisa e Inovação

- Os fundamentos teóricos são cruciais para a pesquisa em ciência da computação.
- Muitas descobertas e inovações tecnológicas atuais, como a criptografia, a inteligência artificial e a computação quântica, têm suas raízes na teoria da computação.

## 6. Resolução de Problemas e Pensamento Crítico

- O estudo dos aspectos teóricos da computação aprimora habilidades de resolução de problemas e pensamento crítico.
- Ele ensina a abordar problemas de maneiras estruturadas e lógicas, habilidades essenciais para qualquer cientista da computação.

## 7. Desenvolvimento de Novas Tecnologias

- Muitas das tecnologias emergentes são baseadas em princípios teóricos da computação.
- A compreensão desses princípios é fundamental para continuar a inovação e o avanço tecnológico.



# O que é Computação Teórica?

- **Definição**

- É um ramo da ciência da computação que se ocupa do estudo dos fundamentos teóricos da informação, computação e algoritmos.
- Ela se preocupa em entender quais problemas podem ser resolvidos com computadores, como eficientemente esses problemas podem ser resolvidos, e quais são os limites do que pode ser computado.

# Quem foi Alan Turing?

- Alan Turing foi um matemático e cientista da computação britânico, amplamente considerado o pai da ciência da computação teórica e da inteligência artificial.
- Suas contribuições durante a Segunda Guerra Mundial, especialmente na quebra do código Enigma, foram cruciais.
- <https://www.youtube.com/watch?v=wFvLFGNLvbU>

# Máquina de Turing - O Conceito

- Em 1936, Turing propôs o conceito de uma "máquina de Turing", um dispositivo abstrato que manipula símbolos em uma tira de fita de acordo com uma tabela de regras.
- Apesar de simples, a máquina de Turing pode simular a lógica de qualquer algoritmo de computador.

# Máquina de Turing - Componentes Chave

- Fita Infinita: Dividida em células, cada uma contendo um símbolo.
- Cabeça de Leitura/Escrita: Lê e escreve símbolos na fita e pode mover-se para a esquerda ou direita.
- Tabela de Regras (ou Função de Transição): Define o que a máquina faz, dependendo do símbolo lido.

# Máquina de Turing - Importância

## 1. Fundamento da Computabilidade:

- A máquina de Turing tornou-se um modelo fundamental para entender o que pode ser computado.
- Ela é a base para a definição de algoritmos e para o estudo da computabilidade e da complexidade computacional.

# Máquina de Turing - Importância

## 2. Teoria da computação

- O conceito de Turing é central na teoria da computação.
- Ele estabeleceu as fundações para a teoria da decisão, mostrando que existem problemas que não podem ser resolvidos por qualquer máquina de Turing (problemas indecidíveis).

# Máquina de Turing - Importância

## 3. Desenvolvimento de Computadores Modernos:

- Embora a máquina de Turing seja um modelo abstrato, ela influenciou o desenvolvimento dos primeiros computadores e continua a ser um ponto de referência na concepção de novos modelos computacionais.

# Máquina de Turing - Importância

## 4. Inteligência artificial

- Turing também propôs o famoso "Teste de Turing" como um critério para avaliar a inteligência de uma máquina, influenciando profundamente o campo da inteligência artificial.



# Máquina de Turing - Vídeos

- Vídeo 1
- [A Turing Machine - Overview.mp4](#)
- Vídeo 2
- [The LEGO Turing Machine.mp4](#)

# Aplicabilidade da Teoria da Computação na Prática

- Criptografia e Segurança:
  - Fundamental para o desenvolvimento de sistemas de criptografia que protegem dados e comunicações em nosso mundo digital.

# Aplicabilidade da Teoria da Computação na Prática

- Desenvolvimento de Software:
  - Aplicada no design de linguagens de programação, compiladores, e sistemas operacionais.

# Aplicabilidade da Teoria da Computação na Prática

- Inteligência Artificial e Aprendizado de Máquina:
  - Conceitos teóricos são aplicados para construir algoritmos inteligentes que podem aprender, adaptar e tomar decisões.

# Aplicabilidade da Teoria da Computação na Prática

- Análise de Dados:
  - Técnicas da teoria da computação são essenciais para processar e analisar grandes volumes de dados (Big Data).

# Aplicabilidade da Teoria da Computação na Prática

- Otimização e Logística:
  - Utilizada em sistemas de otimização que encontram soluções eficientes para problemas logísticos, de produção, e de redes.

# Aplicabilidade da Teoria da Computação na Prática

- Computação Quântica:
  - Fornece o framework teórico para o desenvolvimento da próxima geração de computadores, com potencial para revolucionar diversos campos.

# Aplicações da Teoria da Computação no Mundo Real

- **Criptografia - Exemplo:**
  - A segurança online, como a criptografia de sites de e-commerce e a proteção de transações bancárias online.
- **Teoria Aplicada:**
  - Algoritmos de chave pública (como RSA) baseiam-se em princípios teóricos da matemática e da computação, como a dificuldade de fatorar números grandes.



# Aplicações da Teoria da Computação no Mundo Real

- **Otimização de Buscas na Internet - Exemplo:**
  - Motores de busca como o Google usam algoritmos complexos para classificar páginas web e fornecer resultados relevantes rapidamente.
- **Teoria Aplicada:**
  - Algoritmos de ordenação e busca, e a teoria dos grafos, são essenciais para entender e otimizar esses processos.

# Aplicações da Teoria da Computação no Mundo Real

- **Inteligência Artificial e Aprendizado de Máquina - Exemplo:**
  - Sistemas de recomendação usados por serviços de streaming como Netflix ou plataformas de compras online.
- **Teoria Aplicada:**
  - Conceitos de aprendizado de máquina, redes neurais e otimização estocástica, todos fundamentados na teoria da computação.

# Aplicações da Teoria da Computação no Mundo Real

- **Processamento de Linguagem Natural - Exemplo:**
  - Assistente virtuais como Siri ou Google Assistant interpretando e respondendo a comandos de voz.
- **Teoria Aplicada:**
  - Análise sintática e semântica baseada em teorias de autômatos e linguagens formais.

# Aplicações da Teoria da Computação no Mundo Real

- **Desenvolvimento de Software e Compiladores - Exemplo:**
  - Compiladores que transformam código de linguagem de alto nível em linguagem de máquina.
- **Teoria Aplicada:**
  - Teoria de linguagens formais e autômatos são cruciais para o design e a implementação de compiladores eficientes.

# Aplicações da Teoria da Computação no Mundo Real

- **Sistemas de Banco de Dados - Exemplo:**
  - Bancos de dados que suportam transações online e armazenam enormes quantidades de dados.
- **Teoria Aplicada:**
  - Algoritmos de indexação e busca, teoria de grafos e algoritmos de otimização.

# Aplicações da Teoria da Computação no Mundo Real

- **Segurança de Redes e Sistemas - Exemplo:**
  - Detecção e prevenção de intrusões em sistemas de rede.
- **Teoria Aplicada:**
  - Teoria dos grafos para analisar padrões de tráfego de rede e identificar atividades suspeitas.

# Aplicações da Teoria da Computação no Mundo Real

- **Computação Gráfica e Visualização de Dados - Exemplo:**
  - Efeitos visuais em filmes e jogos, visualização de dados complexos em ciência e engenharia.
- **Teoria Aplicada:**
  - Algoritmos geométricos e teoria de grafos.

# Aplicações da Teoria da Computação no Mundo Real

- **Bioinformática e Modelagem Computacional - Exemplo:**
  - Análise de sequências de DNA e modelagem de estruturas moleculares em biologia.
- **Teoria Aplicada:**
  - Algoritmos de busca e alinhamento de sequências, teoria da complexidade para entender limitações computacionais.



# Estudos de caso – Inteligência artificial

- **AlphaGo da DeepMind:**
  - Um programa de computador que joga o jogo de tabuleiro Go. AlphaGo ganhou notoriedade ao derrotar um campeão mundial em 2016.
- **Conceitos Teóricos:**
  - Redes neurais profundas, algoritmos de busca de árvore, e aprendizado de reforço.
  - Demonstrou o potencial da IA em resolver problemas complexos que requerem intuição e criatividade, desafiando a crença de que tais tarefas são exclusivamente humanas.

# Estudos de caso – Inteligência artificial

- **Sequenciamento de DNA com IA:**
  - Uso de algoritmos de IA para acelerar o sequenciamento do DNA humano.
- **Conceitos Teóricos:**
  - Algoritmos de aprendizado de máquina para análise de grandes conjuntos de dados biológicos.
  - Permite avanços significativos em genética e medicina personalizada, contribuindo para tratamentos mais eficazes e diagnósticos precoces.

# Estudos de caso – Sistemas Operacionais

- **Gerenciamento de Memória no Windows 10:**
  - Sistema operacional amplamente utilizado com sofisticado gerenciamento de memória.
- **Conceitos Teóricos:**
  - Paginação, segmentação e algoritmos de substituição de páginas (como LRU - Least Recently Used).
  - Otimiza o uso da memória do computador, melhorando o desempenho e a eficiência do sistema.

# Estudos de caso – Desenvolvimento de Software

- **Desenvolvimento do Navegador Google Chrome:**
  - Um dos navegadores web mais populares, conhecido por sua velocidade e eficiência.
- **Conceitos Teóricos:**
  - Algoritmos de renderização de página, gerenciamento eficiente de processos e threads, isolamento de processos para segurança.
  - Melhora a experiência do usuário na navegação web, oferecendo segurança e eficiência na utilização de recursos do sistema.

# Breve resumo sobre o que será estudado

- **Gramáticas Formais**
  - São um conjunto de regras para gerar strings em um determinado idioma.
  - Elas são fundamentais na teoria da computação e linguagens formais, desempenhando um papel crucial na definição de linguagens de programação e no design de compiladores.

# Breve resumo sobre o que será estudado

- **Tipos de Gramáticas (Hierarquia de Chomsky):**
  - **Gramáticas Tipo 0 (Reescrita Irrestrita):**
    - Incluem todos os sistemas de regras de reescrita. São capazes de expressar qualquer linguagem computável, incluindo linguagens que não são decidíveis.
  - **Gramáticas Tipo 1 (Sensíveis ao Contexto):**
    - Podem ter regras que dependem de um contexto. São usadas para descrever linguagens mais complexas, como algumas características de linguagens de programação natural.

# Breve resumo sobre o que será estudado

- **Tipos de Gramáticas (Hierarquia de Chomsky):**
  - **Gramáticas Tipo 2 (Livre de Contexto):**
    - Cada regra de produção depende apenas de um único símbolo não-terminal. São amplamente usadas para definir a sintaxe de linguagens de programação.
  - **Gramáticas Tipo 3 (Regulares):**
    - As mais simples; podem ser expressas por expressões regulares e são usadas em design de scanners e algumas formas de análise léxica.

# Breve resumo sobre o que será estudado

- **Gramáticas Formais**

- As gramáticas, especialmente as Livres de Contexto (Tipo 2), são essenciais para definir a sintaxe de linguagens de programação.
- Permitem a construção de compiladores e interpretadores ao fornecer uma estrutura formal que pode ser analisada e processada por máquinas.
- A análise sintática em compiladores, um processo essencial na conversão de código-fonte para código de máquina ou bytecode, baseia-se fortemente em gramáticas formais.



# Breve resumo sobre o que será estudado

- **Autômatos**
  - São modelos matemáticos abstratos para máquinas que executam operações em uma entrada e mudam de estado de acordo com essa entrada.
  - São fundamentais na teoria dos autômatos e linguagens formais.

# Breve resumo sobre o que será estudado

- **Autômatos**

- Autômatos são usados na análise léxica de compiladores para reconhecer tokens.
- Expressões regulares definem padrões para identificar esses tokens.
- Facilitam a construção de analisadores léxicos, tornando a tradução de linguagens de programação mais eficiente.

# Breve resumo sobre o que será estudado

- **Autômatos**
  - Autômatos finitos são aplicados no design de circuitos sequenciais, onde a saída depende tanto das entradas atuais quanto do estado anterior do sistema.
  - Essencial para o design de circuitos digitais, como memórias e unidades de controle em microprocessadores.

# Breve resumo sobre o que será estudado

- **Autômatos**
  - Autômatos modelam protocolos de comunicação, descrevendo as sequências de eventos permitidos durante a comunicação entre dois ou mais sistemas.
  - Garantem a transferência confiável e eficiente de dados em redes de computadores e sistemas de telecomunicações.

# Breve resumo sobre o que será estudado

- **Expressões regulares**
  - São sequências de caracteres que formam um padrão de busca. Elas são usadas para especificar conjuntos de strings de forma concisa e flexível.

# Breve resumo sobre o que será estudado

- **Expressões regulares**
  - Expressões regulares são amplamente utilizadas em ferramentas de processamento de texto e scripts para busca e manipulação de padrões em textos.
  - Permitem a realização de tarefas complexas de processamento de texto, como validação de dados, busca e substituição de informações em grandes volumes de dados.

# Nesta aula vimos

- Como os aspectos teóricos sustentam muitos avanços e inovações tecnológicas.
- Falamos de temas como máquinas de Turing, complexidade computacional, pesquisa e inovação, resolução de problemas e pensamento crítico, entre outros.

# Nesta aula vimos

- Sobre a relevância da teoria da computação em campos como criptografia, desenvolvimento de software, inteligência artificial, análise de dados, otimização e logística, computação quântica, entre outros.
- Exemplos práticos da aplicação de conceitos teóricos em casos reais.
- Breve descrição dos tópicos específicos, como gramáticas formais, tipos de gramáticas, autômatos e expressões regulares.



# Pesquisa

- Pesquise e escreva ao menos uma página sobre as contribuições de **Donald Knuth** na computação.

## Referências desta aula

HOPCROFT, John E.; MOTWANI, Rajeev; ULLMAN, Jeffrey D.  
Introdução a teoria de autômatos, linguagens e computação. Rio de Janeiro: Campus, 2002.

FIM  
Obrigado

Rodrigo