

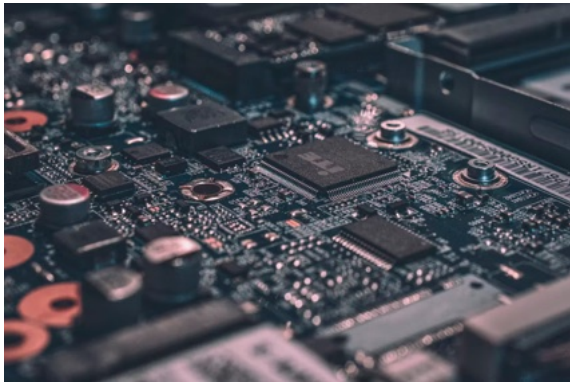
# Sistemas Computacionais

## Prática 0: Introdução

Prof. Dr. Denis M. L. Martins

Ciência de Dados e Inteligência Artificial: 3º Semestre

Imagem: alexkixa @Unsplash



## Objetivo principal

Desenvolver sua capacidade de idealizar e prototipar sistemas computacionais em diferentes níveis de abstração.

- Entender e construir os componentes de um computador moderno.
- Explicar e escrever programas em linguagem assembly.
- Estudar e entender componentes e técnicas de gerenciamento de **sistemas operacionais** modernos.

## Objetivo principal

Desenvolver sua capacidade de idealizar e prototipar sistemas computacionais em diferentes níveis de abstração.

- Entender e construir os componentes de um computador moderno.
- Explicar e escrever programas em linguagem assembly.
- Estudar e entender componentes e técnicas de gerenciamento de **sistemas operacionais** modernos.

- Soberania digital, sustentabilidade.
- Conhecimento técnico nos capacita a usar/construir melhores soluções que atendam aos nossos interesses.
- Exemplo: fim da vida útil de 240 milhões de PCs com Windows 11, além de milhões de outros quando a Apple encerrar o suporte para CPUs Intel.
- Conferir: <https://learn.microsoft.com/pt-br/lifecycle/announcements/windows-11-21h2-end-of-servicing> e [https://eco.kde.org/blog/2024-05-29\\_introducing-ns4nh/](https://eco.kde.org/blog/2024-05-29_introducing-ns4nh/)
- **Alguém** controla seu computador além do seu controle.



**Figura 1:** Campanha "Think Global, Act Local". Imagem: [Karanjot Singh](#).

- Soberania digital, sustentabilidade.
- Conhecimento técnico nos capacita a usar/construir melhores soluções que atendam aos nossos interesses.
- Exemplo: fim da vida útil de 240 milhões de PCs com Windows 11, além de milhões de outros quando a Apple encerrar o suporte para CPUs Intel.
- Conferir: <https://learn.microsoft.com/pt-br/lifecycle/announcements/windows-11-21h2-end-of-servicing> e [https://eco.kde.org/blog/2024-05-29\\_introducing-ns4nh/](https://eco.kde.org/blog/2024-05-29_introducing-ns4nh/)
- **Alguém** controla seu computador além do seu controle.



Figura 1: Campanha "Think Global, Act Local". Imagem: [Karanjot Singh](#).

- Soberania digital, sustentabilidade.
- Conhecimento técnico nos capacita a usar/construir melhores soluções que atendam aos nossos interesses.
- Exemplo: fim da vida útil de 240 milhões de PCs com Windows 11, além de milhões de outros quando a Apple encerrar o suporte para CPUs Intel.
- Conferir: <https://learn.microsoft.com/pt-br/lifecycle/announcements/windows-11-21h2-end-of-servicing> e [https://eco.kde.org/blog/2024-05-29\\_introducing-ns4nh/](https://eco.kde.org/blog/2024-05-29_introducing-ns4nh/)
- **Alguém** controla seu computador além do seu controle.



Figura 1: Campanha "Think Global, Act Local". Imagem: [Karanjot Singh](#).

### Aulas teóricas

A abordagem teórica treina a capacidade de analisar e explicar sistemas de hardware e software.

### Aulas práticas

O aprendizado é consolidado por meio da experimentação e criação de circuitos digitais e técnicas de sistemas operacionais modernos.



Figura 2: Taxonomia de Bloom. Imagem: [Wikipedia](#).

- Dinâmica das aulas:
  - ▶ Resolução de exercícios
  - ▶ Trabalhar em projetos ao longo do semestre<sup>1</sup>
- Critérios de avaliação:
  - ▶ **Correção das Implementações (70%)** – Verificação se os circuitos digitais, simulações e códigos funcionam corretamente, atendendo aos requisitos propostos. Habilidade em identificar e corrigir erros, otimizar soluções e propor alternativas.
  - ▶ **Organização e Clareza do Código/Projeto (15%)** – Estruturação do código, uso adequado de nomenclaturas e documentação dos experimentos realizados.
  - ▶ **Engajamento e Participação (15%)** – Dedicção durante as atividades práticas, contribuição em discussões e trabalho em equipe.

---

<sup>1</sup>Projetos valem nota. Mais detalhes nas próximas semanas, quando tivermos *momentum* ("embalo").



### Digital: design e simulação de circuitos digitais

Link para download: <https://github.com/hneemann/Digital>

Necessário instalar JRE: <https://www.java.com/en/download/>

### VSCode: ambiente de desenvolvimento integrado (IDE)

Link para download: <https://code.visualstudio.com/>

Necessário instalar a extensão necessária MASM/TASM:

<https://marketplace.visualstudio.com/items?itemName=xsro.masm-tasm>

Tutorial de instalação: <https://www.youtube.com/watch?v=pBh29mURXLk>

- N. Nisan, S. Schocken. The Elements of Computing Systems, Building a Modern Computer from First Principles, 2nd ed., MIT, 2005.<sup>2</sup>
- R.J. Tocci, N.S. Widmer, G. Moss. Sistemas digitais: princípios e aplicações, 12ª Ed., Pearson, 2019.<sup>3</sup>
- TANENBAUM, A.; Sistemas Operacionais Modernos. 4a ed. Pearson Brasil, 2015.<sup>4</sup>

---

<sup>2</sup>Também conhecido como Nand2Tetris <https://www.nand2tetris.org/>

<sup>3</sup>Na biblioteca existe a 11ª edição de 2011. Na Pearson digital parece que existe 12ª Ed. 2017

<sup>4</sup> Observação: Na biblioteca existe a 3ª edição de 2010. Saiu uma nova edição em 2024

- N. Nisan, S. Schocken. The Elements of Computing Systems, Building a Modern Computer from First Principles, 2nd ed., MIT, 2005.<sup>5</sup>
- R.J. Tocci, N.S. Widmer, G. Moss. Sistemas digitais: princípios e aplicações, 12ª Ed., Pearson, 2019.<sup>6</sup>
- TANENBAUM, A.; Sistemas Operacionais Modernos. 4a ed. Pearson Brasil, 2015.<sup>7</sup>

---

<sup>5</sup>Também conhecido como Nand2Tetris <https://www.nand2tetris.org/>

<sup>6</sup>Na biblioteca existe a 11ª edição de 2011. Na Pearson digital parece que existe 12ª Ed. 2017

<sup>7</sup> Observação: Na biblioteca existe a 3ª edição de 2010. Saiu uma nova edição em 2024

# Organização do Curso

## Comunicação

- Toda a **comunicação** será centralizada no **Canvas**.
- Todo o material será disponibilizado lá.
- Sempre verifiquem os avisos.
- Usem o fórum de dúvidas.
- Submissão de tarefas.
- Enviem, se necessário, e-mail para:  
[denis.martins@puc-campinas.edu.br](mailto:denis.martins@puc-campinas.edu.br)

Imagem: brianjtromp @Unsplash



# Organização do Curso

## Comunicação

- Toda a **comunicação** será centralizada no **Canvas**.
- Todo o material será disponibilizado lá.
- Sempre verifiquem os avisos.
- Usem o fórum de dúvidas.
- Submissão de tarefas.
- Enviem, se necessário, e-mail para:  
[denis.martins@puc-campinas.edu.br](mailto:denis.martins@puc-campinas.edu.br)

Imagem: briantromp @Unsplash



## Questão 1

Instruções de máquina podem ser representadas por uma sequência de bits (i.e., *word*). Quantas instruções diferentes podem ser codificadas usando apenas 6 bits?

## Questão 1

Instruções de máquina podem ser representadas por uma sequência de bits (i.e., *word*). Quantas instruções diferentes podem ser codificadas usando apenas 6 bits? **Resposta:**  $2^6 = 64$

## Questão 1

Instruções de máquina podem ser representadas por uma sequência de bits (i.e., *word*). Quantas instruções diferentes podem ser codificadas usando apenas 6 bits? **Resposta:**  $2^6 = 64$

## Questão 2

Suponha que você queira listar todos os valores possíveis que representam instruções de máquina da tarefa anterior. Em que ordem você os escreveria sistematicamente (ou seja, usando um método simples que garanta que nenhuma instrução seja omitida)?



## Questão 1

Instruções de máquina podem ser representadas por uma sequência de bits (i.e., *word*). Quantas instruções diferentes podem ser codificadas usando apenas 6 bits? **Resposta:**  $2^6 = 64$

## Questão 2

Suponha que você queira listar todos os valores possíveis que representam instruções de máquina da tarefa anterior. Em que ordem você os escreveria sistematicamente (ou seja, usando um método simples que garanta que nenhuma instrução seja omitida)? **Resposta** (proposta): Em ordem crescente (contagem binária).

## Questão 1

Instruções de máquina podem ser representadas por uma sequência de bits (i.e., *word*). Quantas instruções diferentes podem ser codificadas usando apenas 6 bits? **Resposta:**  $2^6 = 64$

## Questão 2

Suponha que você queira listar todos os valores possíveis que representam instruções de máquina da tarefa anterior. Em que ordem você os escreveria sistematicamente (ou seja, usando um método simples que garanta que nenhuma instrução seja omitida)? **Resposta** (proposta): Em ordem crescente (contagem binária).

## Questão 3

Suponha que você precise selecionar entre 32 valores de entrada. Quantos bits seletores seriam apropriados?

## Questão 1

Instruções de máquina podem ser representadas por uma sequência de bits (i.e., *word*). Quantas instruções diferentes podem ser codificadas usando apenas 6 bits? **Resposta:**  $2^6 = 64$

## Questão 2

Suponha que você queira listar todos os valores possíveis que representam instruções de máquina da tarefa anterior. Em que ordem você os escreveria sistematicamente (ou seja, usando um método simples que garanta que nenhuma instrução seja omitida)? **Resposta** (proposta): Em ordem crescente (contagem binária).

## Questão 3

Suponha que você precise selecionar entre 32 valores de entrada. Quantos bits seletores seriam apropriados? **Resposta:** 5 bits.  $2^5 = 32$

# Dúvidas e Discussão

Prof. Dr. Denis M. L. Martins

[denis.martins@puc-campinas.edu.br](mailto:denis.martins@puc-campinas.edu.br)