# Arquitetura de Computadores I

2025/2

Nicolas Ramos Carreira

## Sumário

1	Imp	portância da matéria	2
2	O q	que aprenderemos	
	2.1	Computadores são burros para fazer cálculos	9
		2.1.1 A conta: $(43,1-43,2)+1$	٠
		2.1.2 Propriedades matemáticas	٠
	2.2	Como o Software roda no Hardware	٠
	2.3	A memória	٠
		2.3.1 O que é de fato?	٠
		2.3.2 Bugs no referenciamento de memória são perniciosos .	2
		2.3.3 O porquê entender	4
	2.4		4
3	Inti	rodução	(
	3.1	Acerca da história	(
	3.2		(
	3.3		
	3.4		

### Capítulo 1

## Importância da matéria

Esta é sem dúvidas uma das disciplinas mais importantes para um Cientista da Computação, pois sem ela:

- Não saberemos como o computador funciona de fato
- Não seremos programadores tão bons como podemos ser
- Seremos mais sucessetiveis a cometer determinados erros

### Capítulo 2

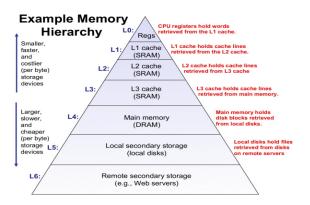
## O que aprenderemos

- 2.1 Computadores são burros para fazer cálculos
- 2.1.1 A conta: (43,1-43,2)+1
- 2.1.2 Propriedades matemáticas
- 2.2 Como o Software roda no Hardware
- 2.3 A memória
- 2.3.1 O que é de fato?

Muitos programas dependem basicamente da memória RAM, mas a memória RAM não existe, é uma abstração limitada. O que existe é um complexo sistema hierárquico de memórias diferentes.

Um exemplo disso é que temos a memória "Registradora", que se encontra no topo da hierarquia. Ela é a memória mais rápida que existe e que roda na mesma velocidade da CPU (ela fica dentro da CPU).

Além disso, temos as memórias cach, que dão suporte a memória registradora.



## 2.3.2 Bugs no referenciamento de memória são perniciosos

Quando estamos fazendo nosso programa, poderemos nos deparar com bugs de memória. Esses bugs podem ser bastante trabalhosos e chatos de lidar. Isso porque são bugs dificeis de serem debuggados

#### 2.3.3 O porquê entender

Dado o contexto anterior, para ser um bom programador, você precisará entender a representação em nível de máquina, na memória, das estruturas de dados e como elas funcionam, pois isso faz uma grande diferença na sua habilidade de evitar e lidar com problemas de referenciamento de memória e vulnerabilidades no seu programa.

Sendo assim, precisaremos entender:

- A hierarquia de memória
- Como a arquitetura da memória e linguagens como C podem levar a bugs de referenciamento de memória que são complicados de debugar e que podem ser distantes do tempo e espaço
- Que a performance da memória não é uniforme e que é necessário otimizar para o baixo nível também

#### 2.4 Insight sobre abstrações

Boa parte do que sabemos sobre os computadores são, na verdade abstrações. No mais baixo nível possível, para o computador realizar uma soma, ele move eletrons.

Sabendo disso, ao longo do tempo, nós fomos precisando de abstrações para representar isso. Foi então que vieram os componentes eletronicos. Porém nós ainda não conseguimos programar/interagir com isso. Dessa forma, abstrairam mais e criaram os circuitos elétricos analogicos. Ainda assim, não conseguimos programar/interagir bem com isso, então criaram a abstração dos circuitos digitais (0, 1, portas lógicas).

## Capítulo 3

## Introdução

- 3.1 Acerca da história
- 3.2 Discussão sobre performance
- 3.3 Falando sobre software
- 3.4 Arquitetura de von neumann