

# EDCO4B

# ESTRUTURAS DE DADOS 2

Aula 09 - Gerenciando Arquivos  
com Registros

Prof. Rafael G. Mantovani

# Licença

Este trabalho está licenciado com uma Licença CC BY-NC-ND 4.0:



maiores informações:

[https://creativecommons.org/licenses/by-nc-nd/4.0/deed.pt\\_BR](https://creativecommons.org/licenses/by-nc-nd/4.0/deed.pt_BR)

# Roteiro

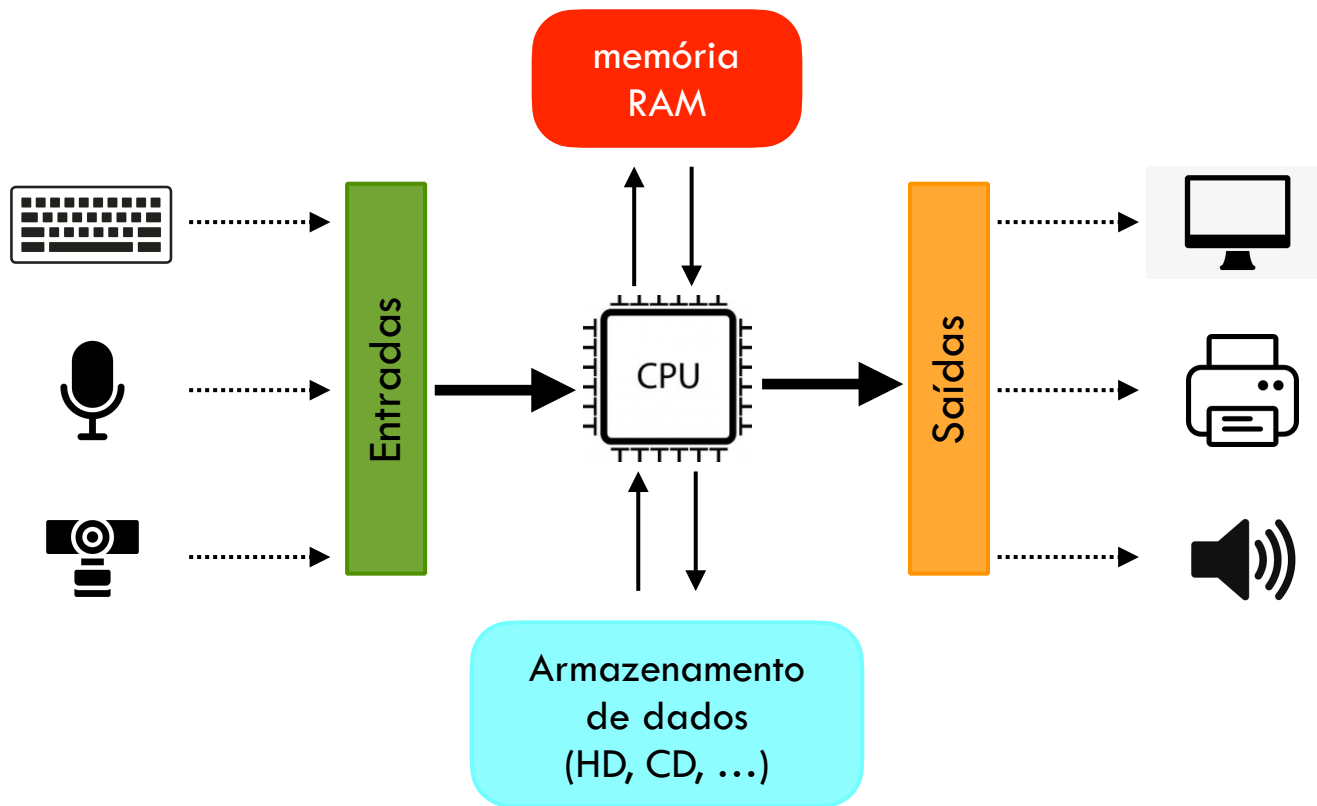


- 1** Introdução
- 2** Acessos a Registros
- 3** Busca Sequencial
- 4** Acesso Direto
- 5** Revisão
- 6** Referências

# Roteiro

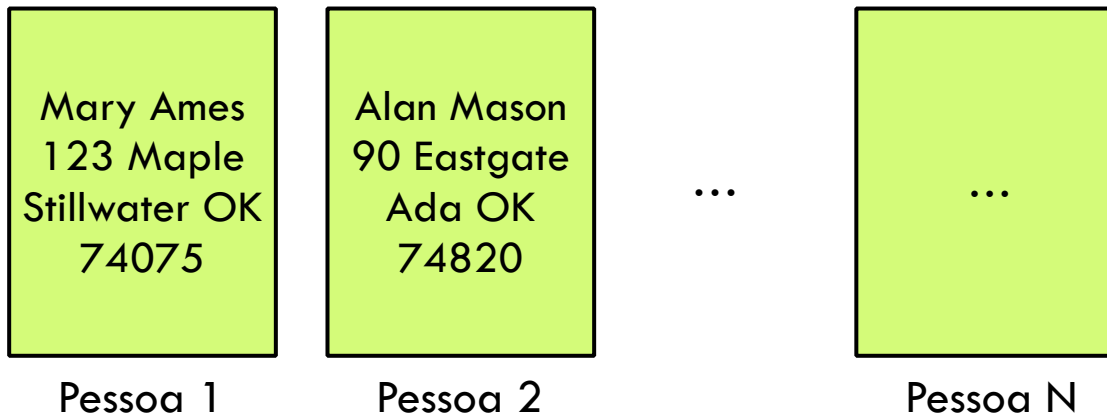
- 1** Introdução
- 2** Acessos a Registros
- 3** Busca Sequencial
- 4** Acesso Direto
- 5** Revisão
- 6** Referências

# Introdução



# Introdução

- **Info:** coleção de nomes e endereços

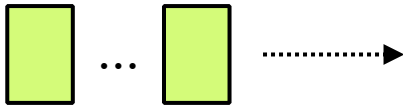


# Roteiro

- 1 Introdução
- 2 Acessos a Registros
- 3 Busca Sequencial
- 4 Cabeçalhos
- 5 Revisão
- 6 Referências

# Acesso a Registros

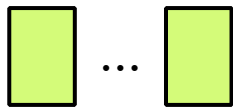
- **Info:** coleção de nomes e endereços





# Acesso a Registros

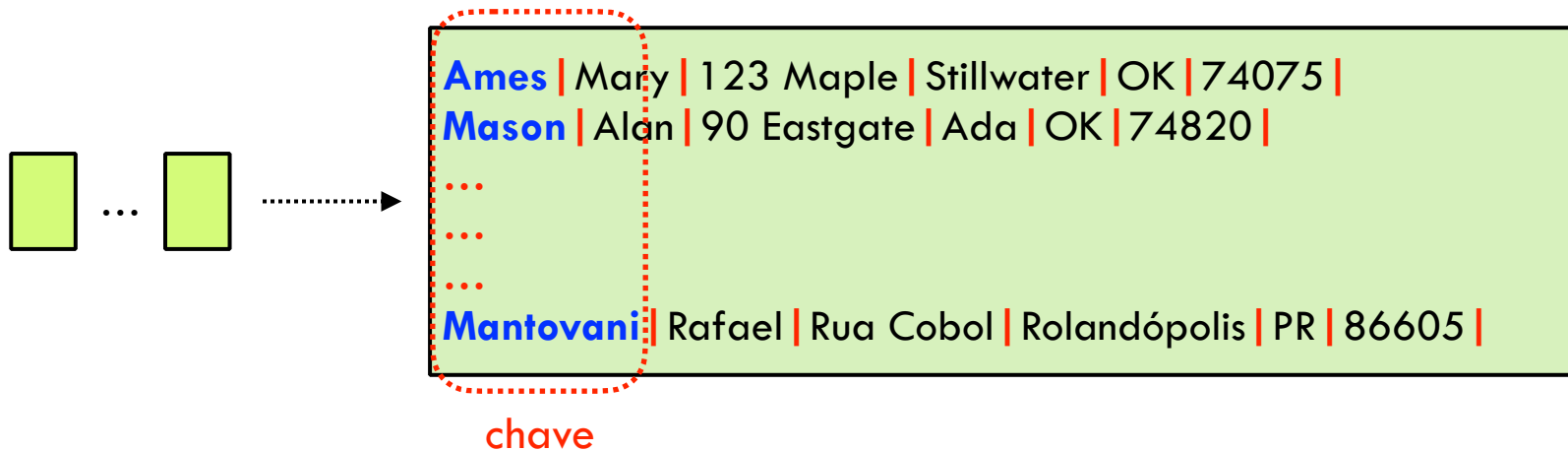
- **Info:** coleção de nomes e endereços



Ames | Mary | 123 Maple | Stillwater | OK | 74075 |  
Mason | Alan | 90 Eastgate | Ada | OK | 74820 |  
...  
...  
...  
Mantovani | Rafael | Rua Cobol | Rolandópolis | PR | 86605 |

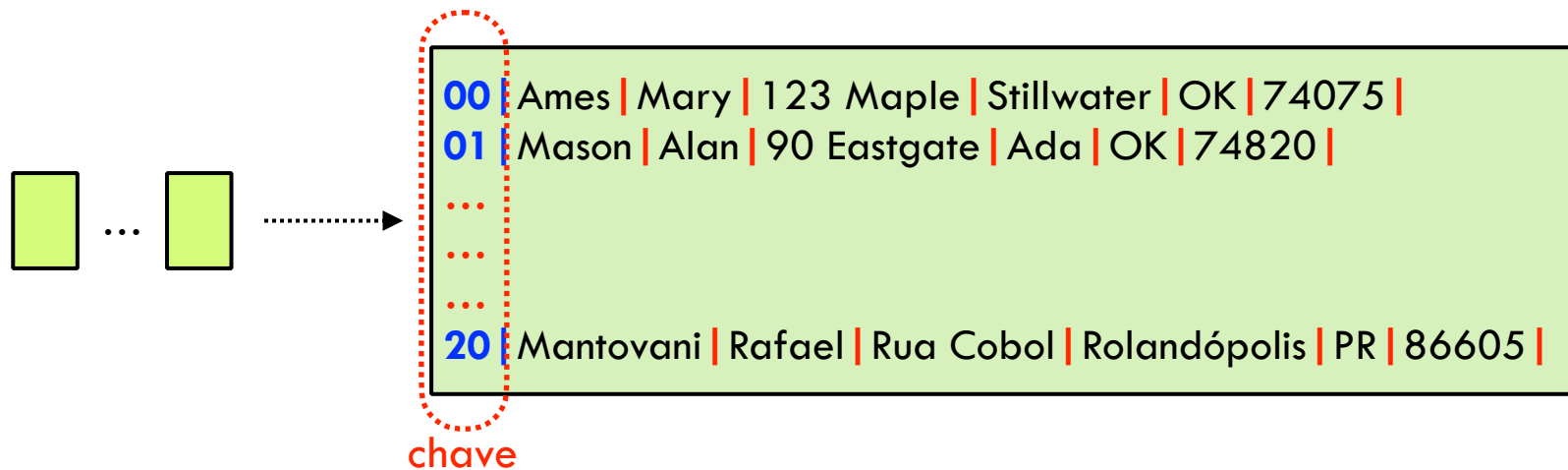
# Acesso a Registros

- **Info:** coleção de nomes e endereços



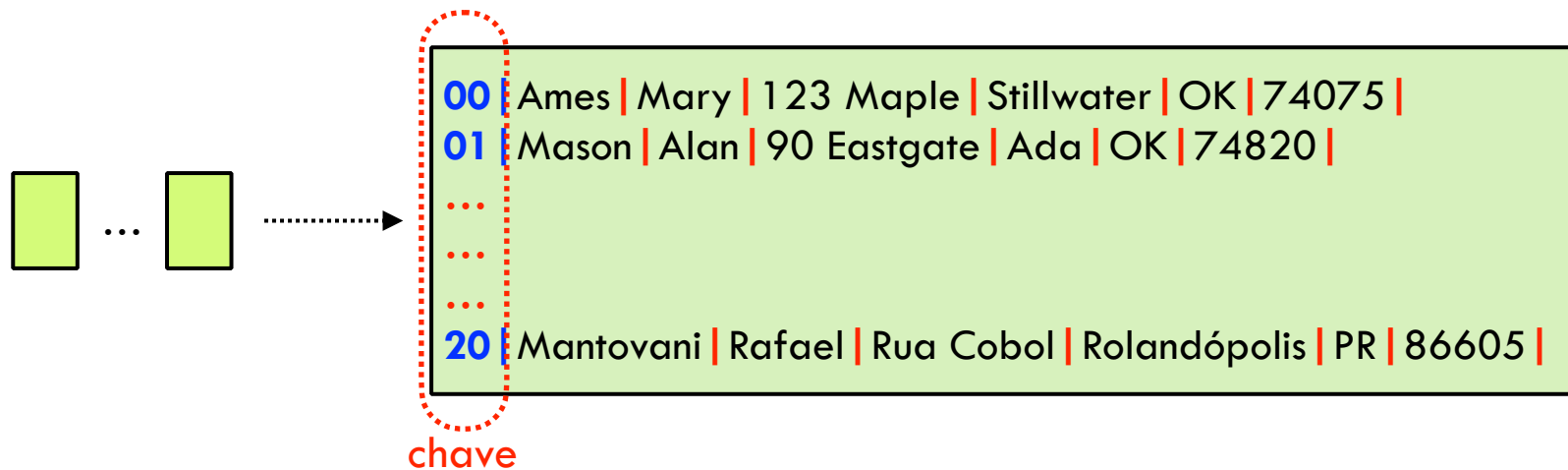
# Acesso a Registros

- **Info:** coleção de nomes e endereços



# Acesso a Registros

- **Info:** coleção de nomes e endereços



É conveniente identificar um registro por uma chave **única**  
(pode-se basear no próprio conteúdo do registro)

# Acesso a Registros

- Chaves devem seguir um formato padrão
  - + regras e procedimentos adequados para converter chaves para o formato padrão
- **Forma Canônica**
  - significa “conforme a regra”
  - é uma representação única

# Acesso a Registros

## □ Forma Canônica

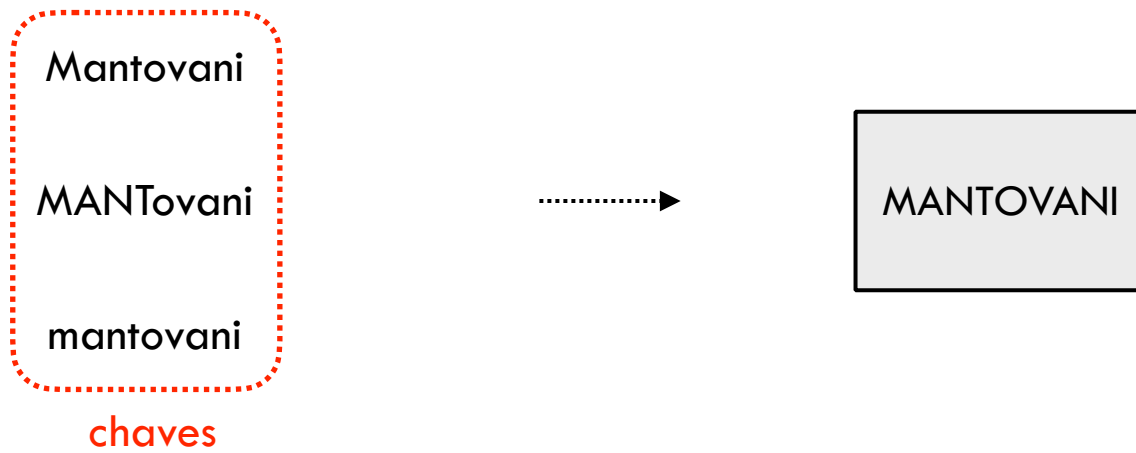
Mantovani  
MANTovani  
mantovani

chaves



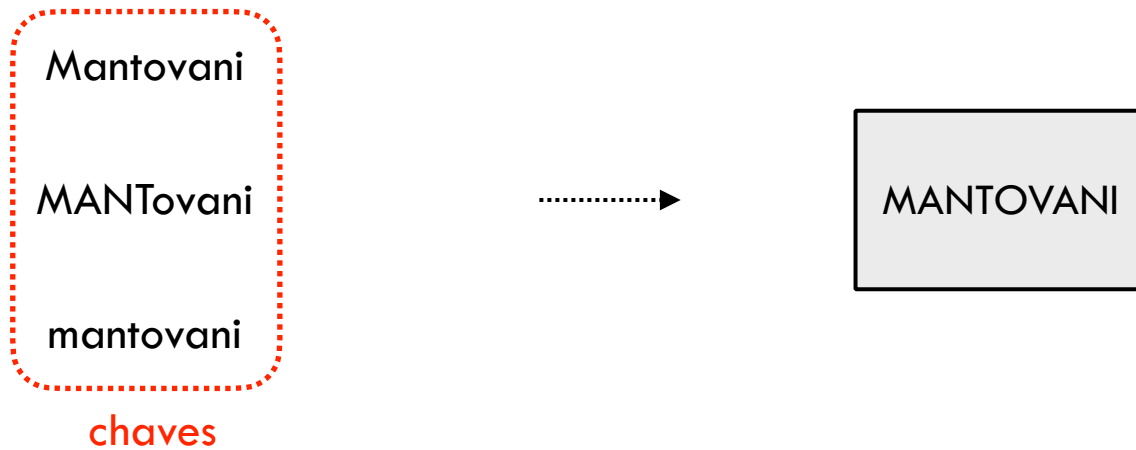
# Acesso a Registros

## □ Forma Canônica



# Acesso a Registros

## □ Forma Canônica



Chaves compostas apenas por letras maiúsculas, sem espaços extras



# Acesso a Registros

## □ Forma Canônica

Ames | Mary | 123 Maple | Stillwater | OK | 74075 |

Mason | Alan | 90 Eastgate | Ada | OK | 74820 |

...

...

...

Mantovani | Rafael | Rua Cobol | Rolandópolis | PR | 86605 |

**AMES** | Mary | 123 Maple | Stillwater | OK | 74075 |

**MASON** | Alan | 90 Eastgate | Ada | OK | 74820 |

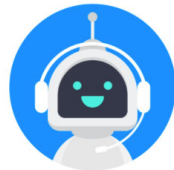
...

...

...

**MANTOVANI** | Rafael | Rua Cobol | Rolandópolis | PR | 86605 |

# Acesso a Registros



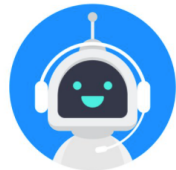
**Usuário**

Adicionar um novo registro



**Arquivo**

# Acesso a Registros



**Usuário**

Adicionar um novo registro

evitar confusões com  
as chaves já existentes



**Arquivo**

# Acesso a Registros



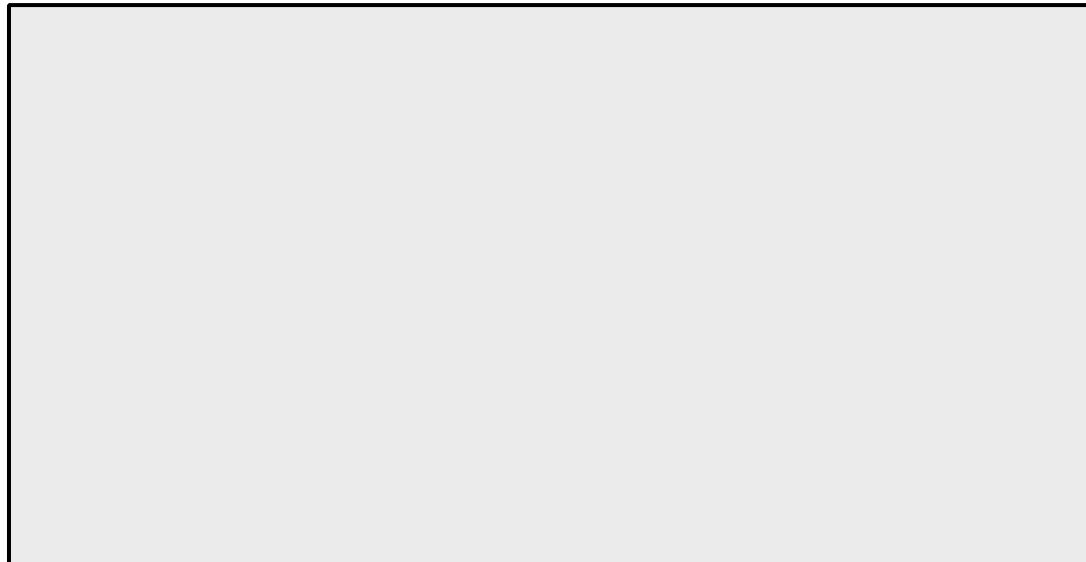
**Usuário**

Adicionar um novo registro

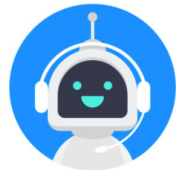
evitar confusões com  
as chaves já existentes



**Arquivo**



# Acesso a Registros



**Usuário**

Adicionar um novo registro

evitar confusões com  
as chaves já existentes

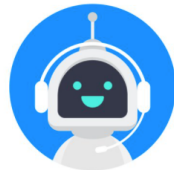


**Arquivo**

**1**

USUARIO | adiciona | novo | registro | .....

# Acesso a Registros



**Usuário**

Adicionar um novo registro

evitar confusões com  
as chaves já existentes



**Arquivo**

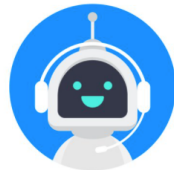
**1**

USUARIO | adiciona | novo | registro | .....

**2**

CRIA | chave | canônica | única | .....

# Acesso a Registros



**Usuário**

Adicionar um novo registro

evitar confusões com  
as chaves já existentes



**Arquivo**

**1**

USUARIO | adiciona | novo | registro | .....

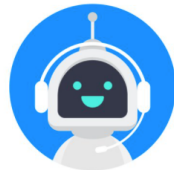
**2**

CRIA | chave | canônica | única | .....

**3**

PROCURA | no | arquivo | se | já | existe | chave | igual | .....

# Acesso a Registros



**Usuário**

Adicionar um novo registro

evitar confusões com  
as chaves já existentes



**Arquivo**

**1**

USUARIO | adiciona | novo | registro | .....

**2**

CRIA | chave | canônica | única | .....

**3**

PROCURA | no | arquivo | se | já | existe | chave | igual | .....

**4a**

SE | não | existe | adiciona | .....

**4b**

SENAO | refaz | a | chave | e | repete | .....



# Acesso a Registros



chaves



- \* conceito de **único** se aplica apenas para chaves **primárias**
- \* é possível ter chaves secundárias (*spoilers*)
- \* chaves secundárias não possuem valores únicos para um registro

# Roteiro



- 1 Introdução
- 2 Acessos a Registros
- 3 Busca Sequencial
- 4 Acesso Direto
- 5 Revisão
- 6 Referências

# Busca Sequencial



**Arquivo**

# Busca Sequencial



**Arquivo**



varredura em  
todos os registros  
(olhando as chaves)

# Busca Sequencial



**Arquivo**



varredura em  
todos os registros  
(olhando as chaves)

simples

# Busca Sequencial



**Arquivo**



varredura em  
todos os registros  
(olhando as chaves)

simples

fácil de  
implementar

# Busca Sequencial



**Arquivo**



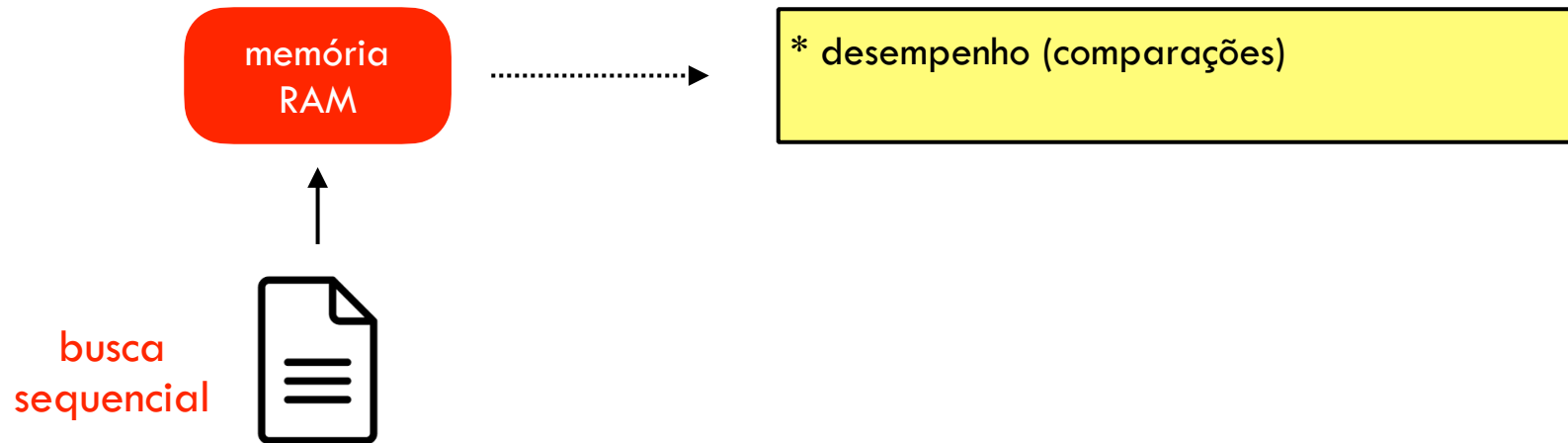
varredura em  
todos os registros  
(olhando as chaves)

simples

fácil de  
implementar

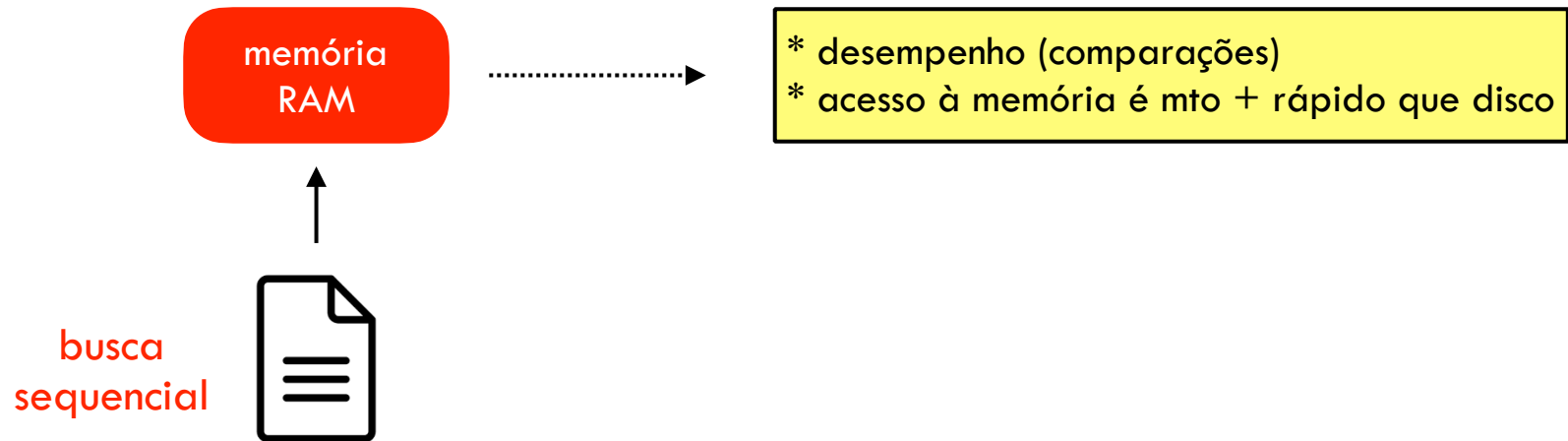
*baseline*

# Busca Sequencial

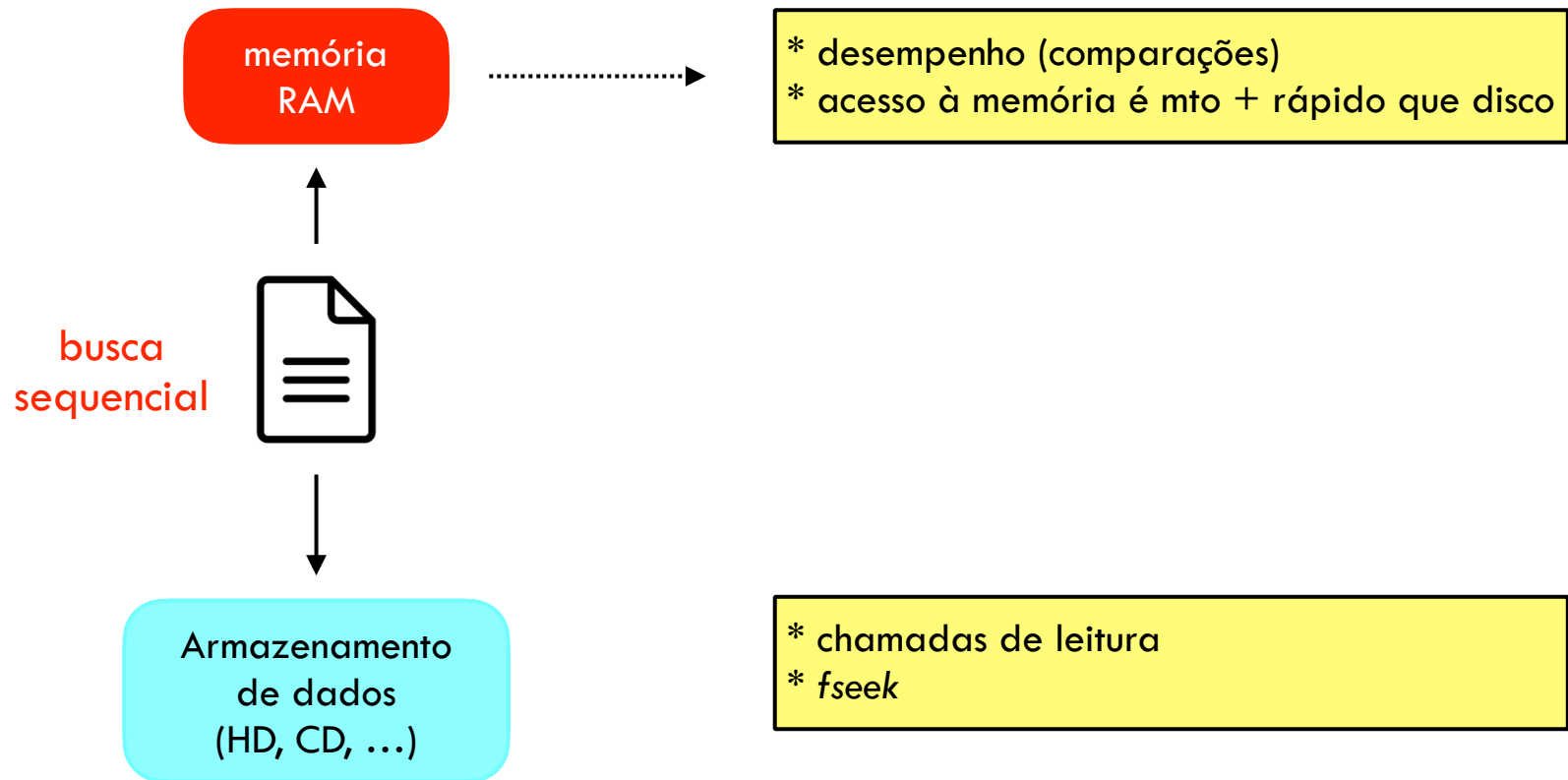




# Busca Sequencial



# Busca Sequencial



# Busca Sequencial

- **Custos** (arquivo com 1000 registros) ?



# Busca Sequencial

- **Custos** (arquivo com 1000 registros) ?



1ª chave  
(1 leitura)

# Busca Sequencial

- **Custos** (arquivo com 1000 registros) ?



1ª chave  
(1 leitura)



última chave  
(1000 leituras)

# Busca Sequencial

- **Custos** (arquivo com 1000 registros) ?

Complexidade:  $O(N)$

duplicar o tamanho do arquivo → duplica custos/tempo



1ª chave  
(1 leitura)



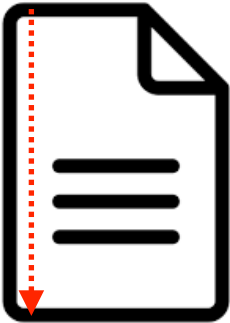
última chave  
(1000 leituras)



média  
(500 leituras)

# Busca Sequencial

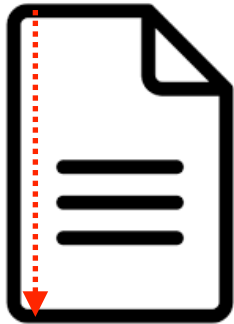
- Quando é uma boa solução?



Busca Sequencial

# Busca Sequencial

- Quando é uma boa solução?



Busca Sequencial

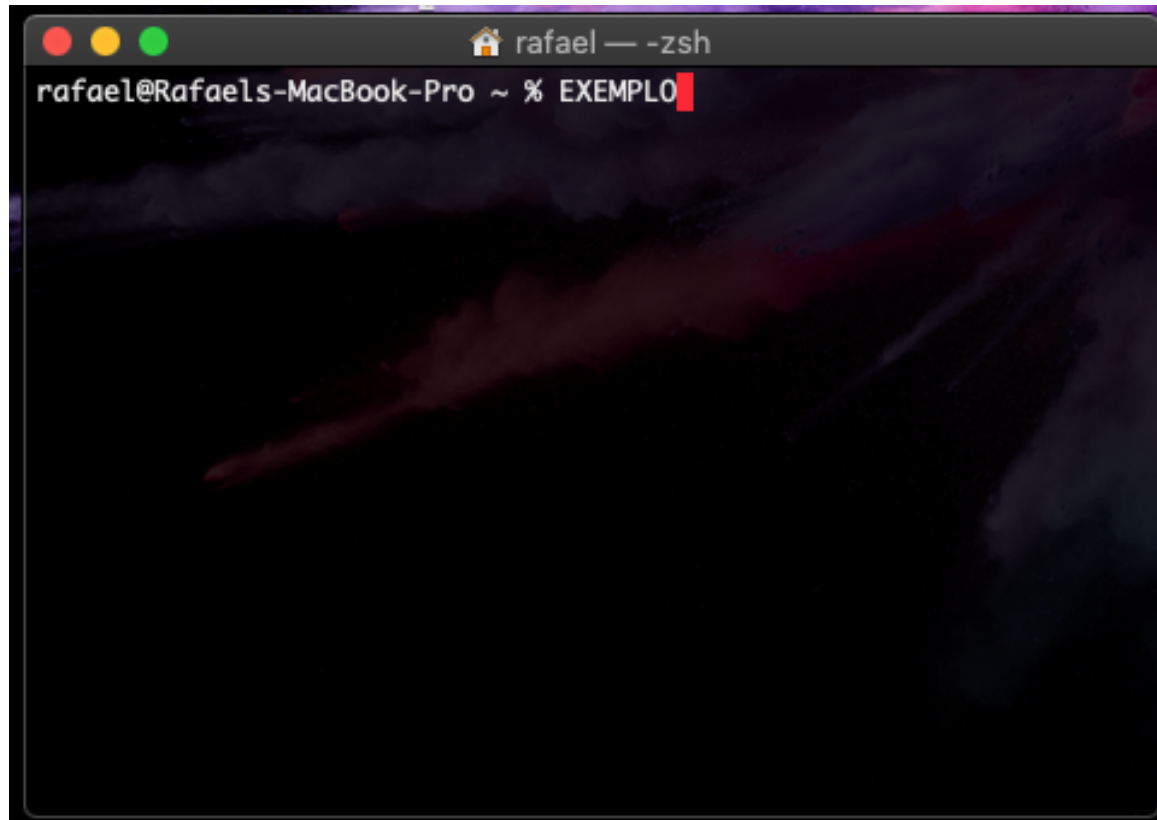
- arquivos ASCII aos quais se procura por algum padrão (**grep**)
- arquivos com poucos registros
- arquivos que dificilmente vão precisar ser consultados
- arquivos com chaves de busca secundárias, cujo número de retornos seja grande



# Busca Sequencial

- Estrutura de arquivos mais comum em sistemas UNIX é:
  - arquivos ASCII com `\n` como delimitador de registro, e espaços em branco como delimitadores de campos
  - white-space/new-line structure
- **cat** (*concatenate*)
- **wc** (*word count*)
- **grep** (*generalised regular expression*)

# Busca Sequencial



# Roteiro

- 1 Introdução
- 2 Acessos a Registros
- 3 Busca Sequencial
- 4 Acesso Direto
- 5 Revisão
- 6 Referências

# Acesso Direto

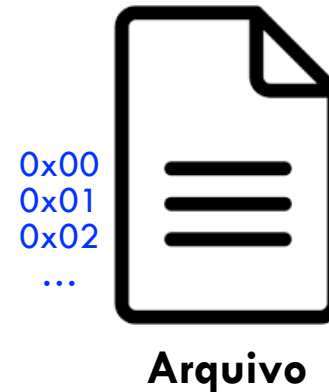
- Opção mais radical/contrária à Busca Sequencial
  - procurar diretamente desde o começo do arquivo um registro e lê-lo
  - usar o endereço de byte do registro como uma referência para o conteúdo

# Acesso Direto

- Opção mais radical/contrária à Busca Sequencial
  - procurar diretamente desde o começo do arquivo um registro e lê-lo
  - usar o endereço de byte do registro como uma referência para o conteúdo



**Endereço?**



# Acesso Direto

- Opção mais radical/contrária à Busca Sequencial
  - procurar diretamente desde o começo do arquivo um registro e lê-lo
  - usar o endereço de byte do registro como uma referência para o conteúdo



**0x01**

**Endereço?**

0x00  
0x01  
0x02  
...



**Arquivo**

# Acesso Direto

- Opção mais radical/contrária à Busca Sequencial
  - procurar diretamente desde o começo do arquivo um registro e lê-lo
  - usar o endereço de byte do registro como uma referência para o conteúdo



# Acesso Direto

- Opção mais radical/contrária à Busca Sequencial
  - procurar diretamente desde o começo do arquivo um registro e lê-lo
  - usar o endereço como uma referência para o conteúdo

\* **Busca Sequencial:**  $O(N)$   
\* **Acesso Direto:**  $O(1)$





# Acesso Direto



- **Problema:** saber como encontrar o registro necessário?

# Acesso Direto

- **Problema:** saber como encontrar o registro necessário?
  - Número Relativo de Registro (*Relative Record Number - RRN*)
  - Arquivos são uma coleção de registros
  - Primeiro registro tem RRN 0, o próximo RRN1, ...
  - Necessitamos de **registros de tamanho fixo**

# Acesso Direto

- Se todos os registros tem o mesmo tamanho, podemos usar o RRN do registro para calcular o **byte offset** do início da informação
  - Valor relativo ao início do arquivo
  - **Exemplo:**

registro RRN **546**

Registros de **128**  
bytes

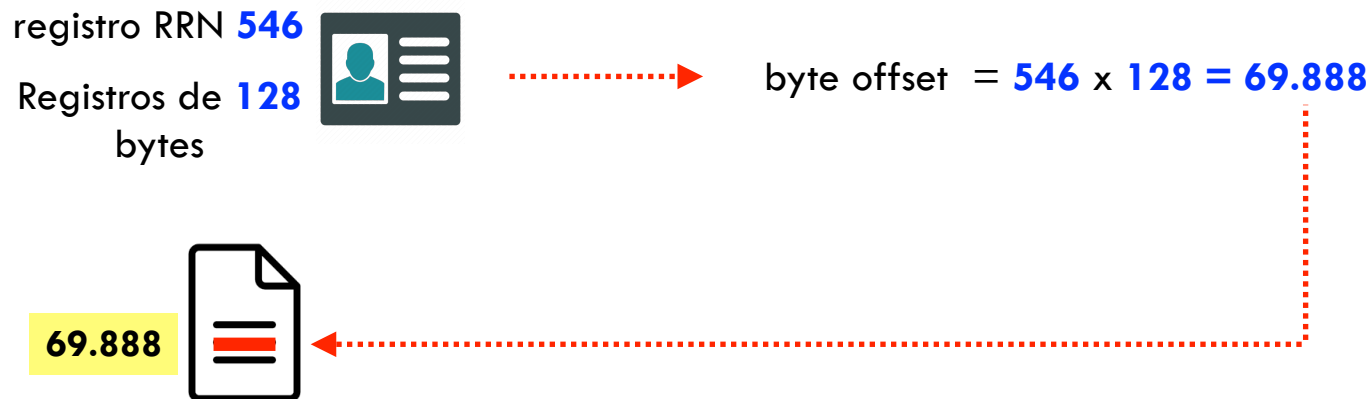


byte offset = **546** x **128** = **69.888**



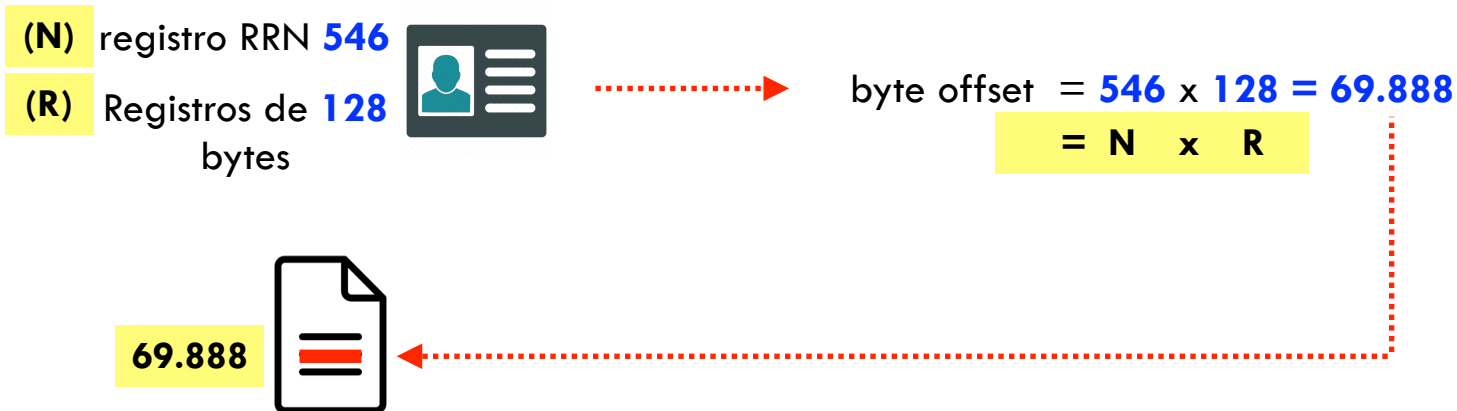
# Acesso Direto

- Se todos os registros tem o mesmo tamanho, podemos usar o RRN do registro para calcular o **byte offset** do início da informação
  - Valor relativo ao início do arquivo
  - **Exemplo:**



# Acesso Direto

- Se todos os registros tem o mesmo tamanho, podemos usar o RRN do registro para calcular o **byte offset** do início da informação
  - Valor relativo ao início do arquivo
  - **Exemplo:**



# Cabeçalhos

- Manter informações adicionais sobre o arquivo para usos futuros
- registro de cabeçalho (***header record***) colocado no começo do arquivo para armazenar essas informações
  - quantidade de registros no arquivo
  - tamanho dos registros
  - data e horário da última edição do arquivo
  - nome do arquivo
  - ...
- registros de cabeçalho são altamente usados na prática

# Exercícios

1) Implemente uma função que simule o comando **grep** do Unix. A função receberá dois parâmetros:

- **um arquivo** texto com registros codificados usando `\n` como delimitador de registros, e `|` como delimitador de campos;
- **uma string** de consulta que deseja-se verificar sua existência e ocorrências no arquivo;

A saída da função é um conjunto com todos os índices das linhas onde a informação foi encontrada no arquivo texto.

# Exercícios



2) Crie uma nova função baseada no exercício anterior e retornar agora todos os registros onde há a ocorrência da string consultada.



# Roteiro



- 1 Introdução
- 2 Acessos a Registros
- 3 Busca Sequencial
- 4 Acesso Direto
- 5 Revisão
- 6 Referências

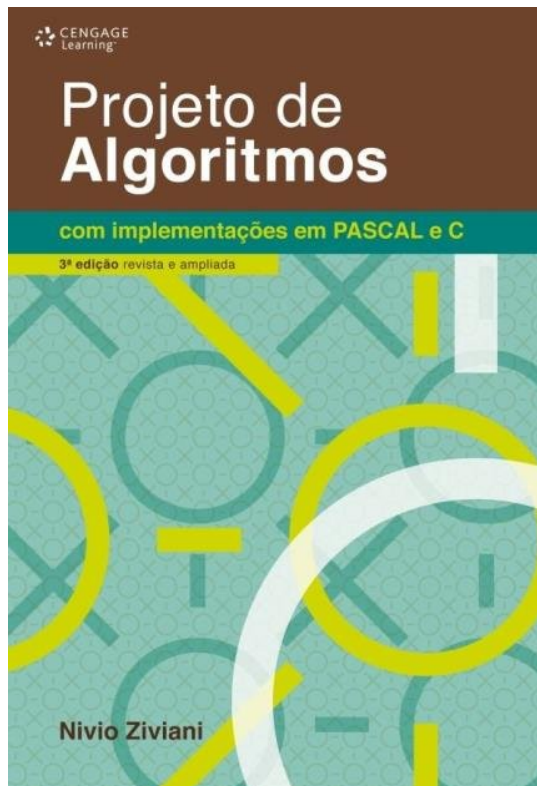
# Revisão

- \* **RRN** (*relative record number*)
- \* Chaves: valores usados para identificar registros (campo)
- \* **Forma Canônica**: chaves únicas, imutáveis e não ambíguas
- \* Busca Sequencial
- \* Acesso Direto
- \* Registros de Cabeçalho

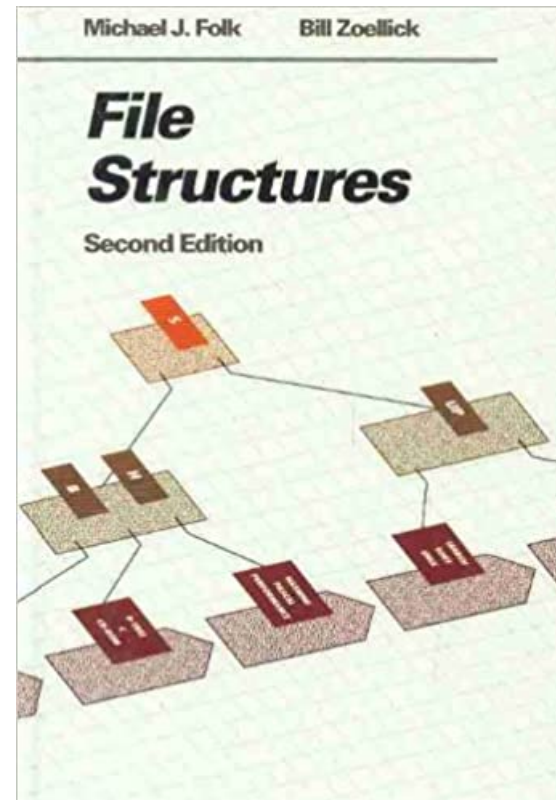
# Roteiro

- 1 Introdução
- 2 Acessos a Registros
- 3 Busca Sequencial
- 4 Acesso Direto
- 5 Revisão
- 6 Referências

# Referências sugeridas

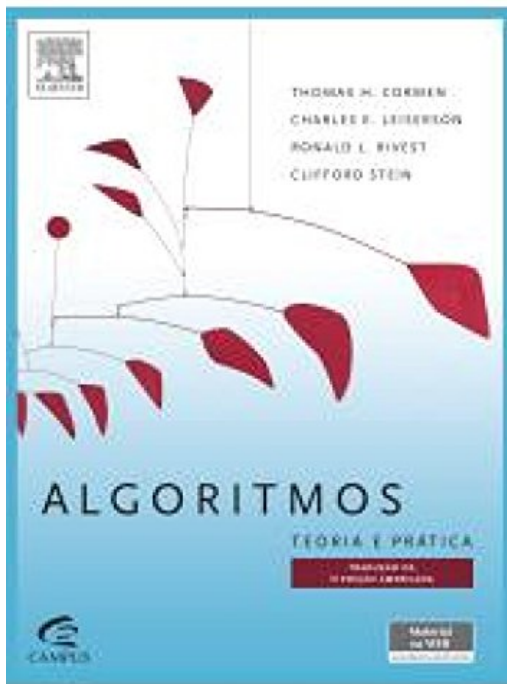


[Ziviani, 2010]



[Folk & Zoellick, 1992]

# Referências sugeridas



[Cormen et al, 2018]



[Drozdek, 2017]

# Perguntas?

Prof. Rafael G. **Mantovani**

[rafaelmantovani@utfpr.edu.br](mailto:rafaelmantovani@utfpr.edu.br)