

# SISTEMAS DE BANCO DE DADOS 1

## AULA 2

### Estudando Arquivos de Dados

Vandor Roberto Vilardi Rissoli



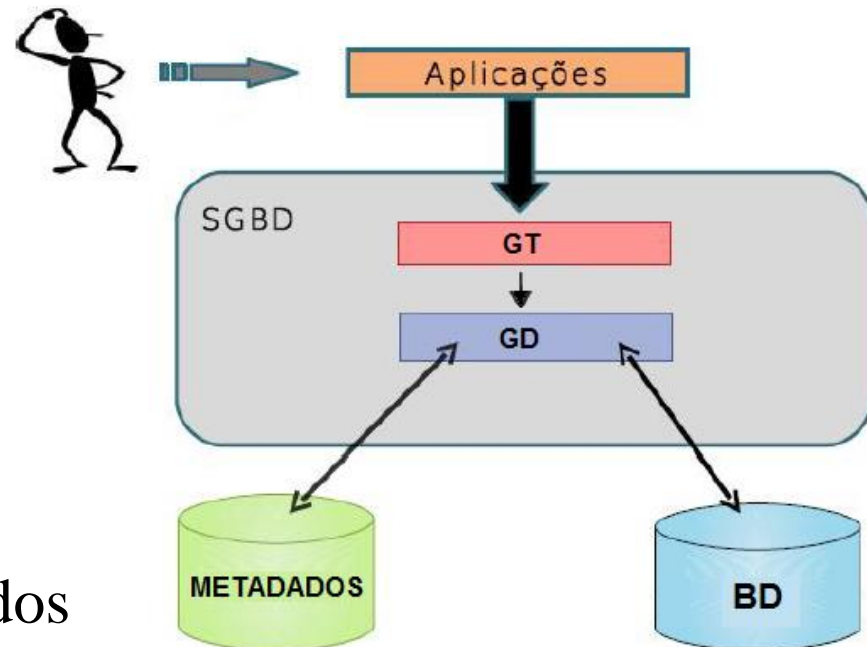
# APRESENTAÇÃO

- Armazenamento de Dados
- Arquivos (alguns tipos principais)
- Operações Algorítmicas sobre Arquivos de Dados
- Referências



# Representação Simplificada de BD

- Aplicações - programas para manipular e consultar dados
- GT (Gerenciador de Transações) - programas responsáveis pelo processamento sobre os dados
- GD (Gerenciador de Dados) - programas responsáveis pelo acesso aos dados
- Metadados - informações sobre os dados
- BD (Banco de Dados) - conjunto de dados armazenados
- SGBD (Sistema Gerenciador de Banco de Dados) - programas responsáveis pelo gerenciamento do Banco de Dados



# Armazenamento de Dados

- **REGISTRO**

- Conjunto de unidades de informação logicamente relacionada;
- Cada unidade de informação corresponde a um campo ou atributo do registro;

- **ARQUIVO**

- Conjunto de registros armazenados em um dispositivo de memória ou armazenamento secundário;
- Recurso de armazenamento persistente (**não volátil**);
- A organização dos registros armazenados dentro de um arquivo pode ser variada e procura distribuir as unidades de informação de maneira mais eficiente ao seu uso.



# Armazenamento de Dados

## Visão Geral dos Meios Físicos de Armazenamento

CACHE: forma de armazenamento mais rápida e cara, sendo pequena e controlada pelo Sistema Operacional (SO) do computador.

Memória PRINCIPAL: mídia de armazenamento para os dados que estão disponíveis ao uso e processamento do computador.

Memória Flash (EEPROM): velocidade próxima a memória Principal, mas o seu armazenamento é mais “complicado” (precisa apagar tudo de uma vez para regravação).



# Armazenamento de Dados

Disco Magnético: mídia primária para o armazenamento de dados por longos períodos, além do acesso on-line (normalmente se tem um BD inteiro armazenado nele).

Disco Óptico: forma mais popular de armazenamento óptico, onde os dados são armazenados opticamente e lidos por laser, podendo ser trocados por outros discos ópticos.

Fita Magnética: dispositivos mais baratos e lentos porque precisam ser acessados sequencialmente a partir do início.

→ As **fitas** são comumente usadas para backup e armazenamento de dados. Elas possuem grande capacidade de armazenamento e podem ser removidas da unidade de fita (troçadas).

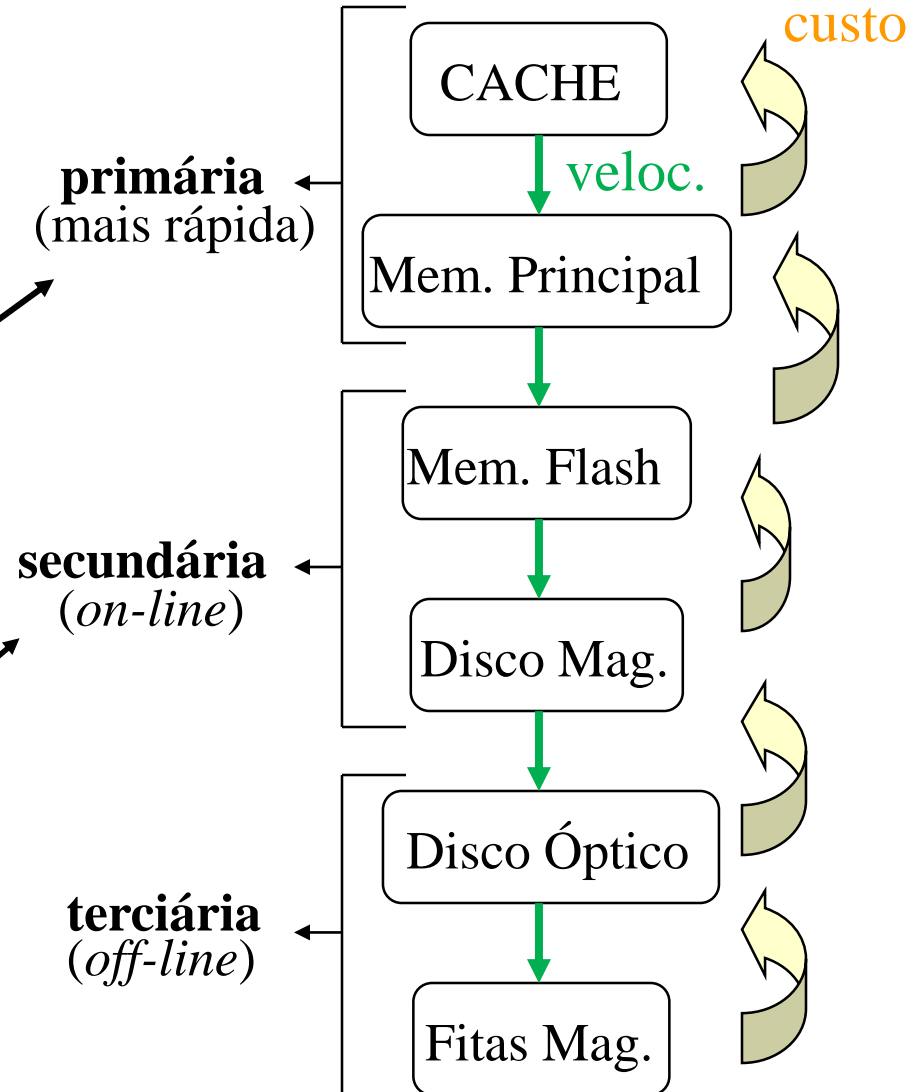
# Armazenamento de Dados

Aspectos importantes na escolha:

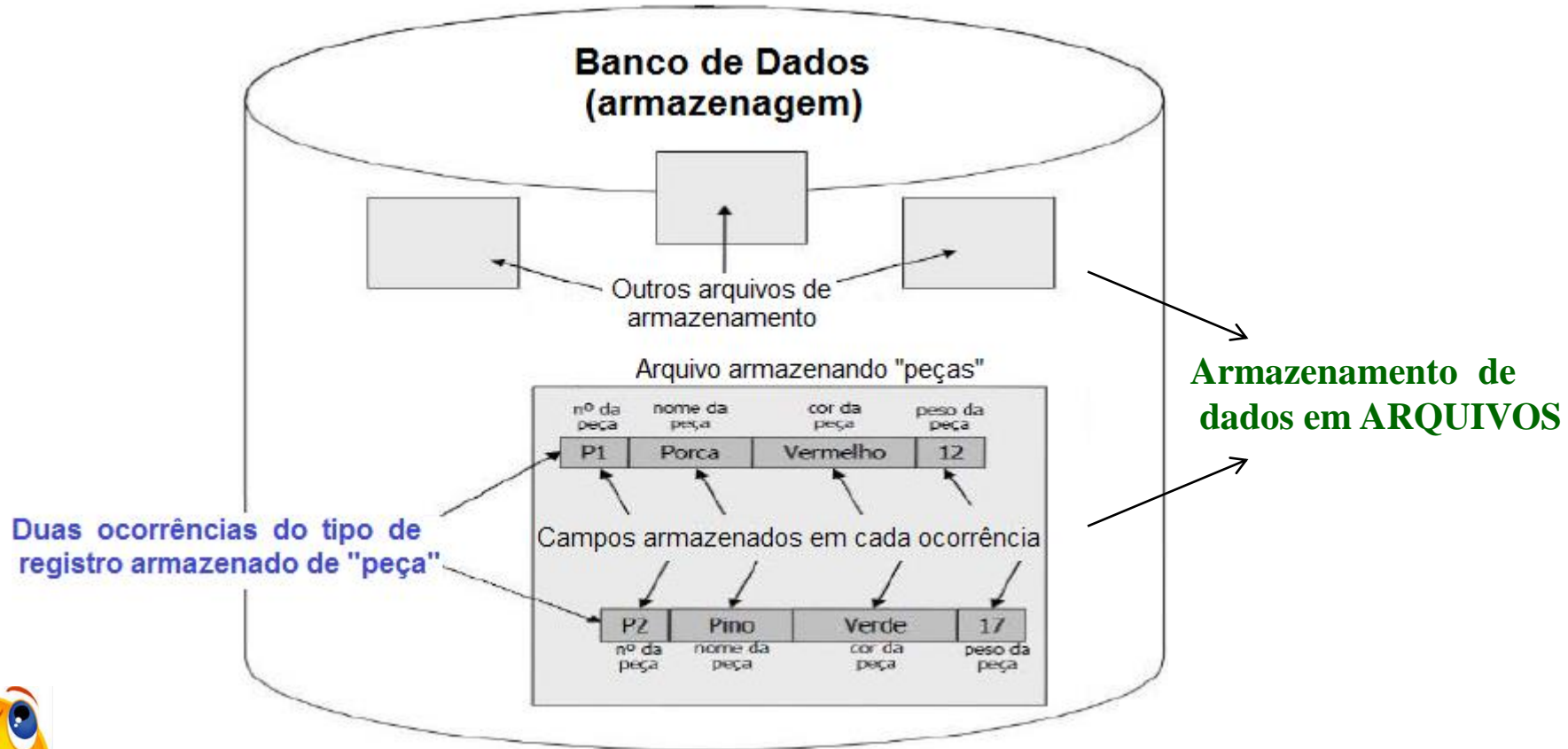
- Custo (\$)
- Velocidade
- Capacidade de armazenamento

Perda de dados com a falta de energia (**volátil**)

Os discos magnéticos fornecem o maior volume de memória **secundária** dos sistemas computacionais modernos.



# Armazenamento de Dados





# ARQUIVOS

- **SEQUENCIAL**

- Registros são dispostos ordenadamente, obedecendo a sequência determinada por uma chave primária, chamada chave de ordenação;
- Acesso aos registros são feitos sequencialmente;
- **Aperfeiçoa** o armazenamento de registros dispostos aleatoriamente no arquivo;
- Existe **perda de flexibilidade** na realização de operações de **modificação** dos dados no arquivo;
- Indicado no acesso de registros usando argumentos que **coincidam com a chave de ordenação** ou em atualizações por lotes (arquivos *batch*).

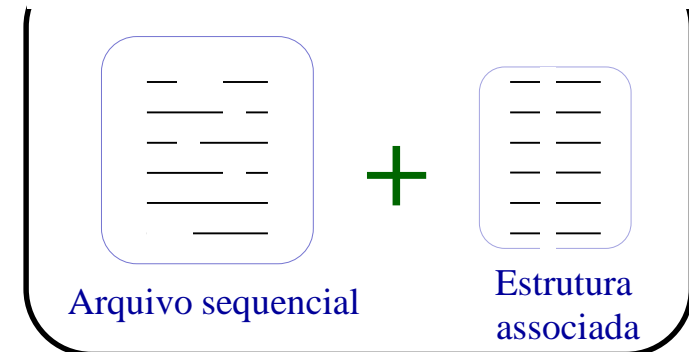


# ARQUIVOS

- **SEQUENCIAL INDEXADO**

- Grande volume de acesso aleatório sobre os arquivos sequencias exige uso de uma estrutura de acesso associada ao arquivo de dados (índice) para torná-lo mais eficiente;
- O **índice** é formado por uma coleção de pares, associando um valor da chave de acesso ao um endereço físico no arquivo, sendo sempre específico a uma chave de acesso;
- Um arquivo sequencial indexado possui **áreas de extensão**, usadas na inserção de novos registros.

## Arquivo Sequencial Indexado



# ARQUIVOS

- **DIRETO**

- Um arquivo direto consiste na instalação dos registros em endereços determinados com base no valor de uma **chave primária**, de modo que se tenha acesso rápido aos registros especificados por argumentos de pesquisa, sem que haja necessidade de percorrer uma estrutura auxiliar (índice);
- Objetivo principal é a eficiência no acesso aleatório, usando a chave do próprio registro ou uma função que calcula o endereço do registro a partir do argumento de pesquisa (*hash*);
- Indicado para uso em aplicações com atualizações arbitrárias.

# ARQUIVOS

## Quadro Comparativo entre Tipos de Arquivos

<u>TIPO DE ARQUIVO</u>	<u>PRINCIPAL VANTAGEM</u>	<u>PRINCIPAL DESVANTAGEM</u>
SEQUENCIAL	Acessos SEQUENCIAIS mais eficientes	Operações de MODIFICAÇÕES mais complexas
SEQUENCIAL INDEXADO	Usam índices que AGILIZAM A CONSULTA e permanecem em memória	Necessidades de ÁREAS DE EXTENSÃO, que ainda precisam ser REORGANIZADAS
DIRETO	Acesso direto, SEM necessidade do ÍNDICE (menor espaço por não possuir mais um arquivo)	Determinar funções que gerem menor número de colisões (consequência do uso de funções não determinísticas para a transformação dos valores da chave de acesso em endereços do arquivo)



→ Existem outros tipos de arquivos, mas estes propiciam um estudo adequado aos objetivos iniciais em Banco de Dados.

# Operações Algorítmicas sobre Arquivo

- Algumas operações básicas sobre arquivos de dados serão representadas de maneira simplificada para o estudo lógico de suas principais operações, sendo elas:
  - **abrir** arquivo;
  - **fechar** arquivo;
  - **recuperar** registro do arquivo;  
(recupera um registro e o armazena na memória);
  - **inserir** novo registro no arquivo;  
(os dados do registro já estão na memória);
  - **alterar** um registro existente no arquivo;
  - **apagar** um registro do arquivo.


# Operações Algorítmicas sobre Arquivo

- A abordagem sobre **REGISTRO** será compreendida como uma Estrutura de Dados Composta Heterogênea, tais como:
  - *struct* em C ou
  - *class* em Java envolvendo somente os atributos.
- A compreensão sobre **ARQUIVO** será entendida como um Recurso Computacional de Armazenamento Secundário de dados que permite alocação de espaço.



# Operações Algorítmicas sobre Arquivo

## Exemplo de Definição de um Arquivo de Dados

- Especificação necessária para definição de um Arquivo que armazenará os dados relacionados ao Registro de uma Pessoa, composto pelos campos:
    - CPF – valor inteiro que guarda o número do CPF;
    - NOME – valor literal (conjunto de caracteres);
    - IDADE – valor inteiro de anos completos.
  - O arquivo será denominado PESSOA.TXT
- 

**arquivo PESSOA.TXT**

CPF	NOME	IDADE

} primeiro registro de pessoa  
 } segundo registro de pessoa



# Operações Algorítmicas sobre Arquivo

## Instruções de Manipulação de Arquivo

declare **registro** <nome do registro>

```
{  <tipo de dado>  nome do elemento;  
    <tipo de dado>  nome de outro elemento;  
}
```

define **arquivo** <nome do arquivo> **sequencial**  
de <nome do registro>;

**abrir** <nome do arquivo> **modo** <tipo de uso do arquivo>;

**fechar** <nome do arquivo>;

O acesso ao arquivo define exatamente o que pode ser realizado em seus dados armazenados, sendo **leitura**, **escrita** ou **ambos** (sem definir o modo de acesso se assume **ambos**).



# Operações Algorítmicas sobre Arquivo

Instrução de leitura de um registro armazenado em um arquivo sequencial para a memória do computador

**recuperar** <nome do arquivo>.<nome do registro>;

- É importante destacar que um arquivo só pode ser lido se já estiver sido aberto com sucesso anteriormente.

Instrução de gravação de um registro em arquivo sequencial a partir dos dados já disponíveis na memória

**inserir** <nome do arquivo>.<nome do registro>;

- O registro deve estar na memória e o arquivo deve estar aberto com modo de **escrita** para ser gravado.



# Operações Algorítmicas sobre Arquivo

Instrução de modificação de um registro já armazenado no arquivo sequencial

**alterar** <nome do arquivo>.<nome do registro>;

- Uma pesquisa localizará o registro armazenado no arquivo e disponibilizará seus dados na memória, sendo estes alterados e gravados novamente nesta posição do arquivo.

Instrução que remove um registro de um arquivo sequencial

**apagar** <nome do arquivo>.<nome do registro>;

- Uma pesquisa localizará o registro armazenado no arquivo, sendo apagado o registro existente nesta posição.

# Operações Algorítmicas sobre Arquivo

Exemplo de algoritmo que cria e armazena dados de empregados em um arquivo sequencial.

```
algoritmo "Gravar dados em arquivo"
```

```
  declare registro FUNCIONÁRIOS
```

```
  {
```

```
    inteiro matriculaFuncional;
```

```
    literal nome;
```

```
    real    salário;
```

```
  }
```

```
  define arquivo EMPREGADOS sequencial de FUNCIONÁRIOS;
```

```
  abrir EMPREGADOS modo escrita;
```

```
  enquanto (novoRegistro = "sim") faça
```

```
    leia FUNCIONÁRIOS;    // le todos os campos de 1 registro em memória
```

```
    inserir EMPREGADOS.FUNCIONÁRIOS; // grava 1 registro no arquivo
```

```
  fimEnquanto;
```

```
  fechar EMPREGADOS;
```

```
fimAlgoritmo
```

# Operações Algorítmicas sobre Arquivo

## CHAVE em Arquivo Direto

- Um campo do registro será escolhido como CHAVE;
  - Campo chave é diferenciado, pois tem a responsabilidade de identificar um único registro no arquivo;
- Para acessar um registro em arquivo direto a chave será informada para localização direta do registro;
- O mecanismo de gerência do arquivo direto é capaz de associar a chave ao local físico do registro procurado;
- A chave identifica um único registro no arquivo
- A variável EOF (*End Of File*) está associada a um arquivo, sendo verdadeira (*true*) quando não houver mais registros armazenados (chegou no fim do arquivo).

# Operações Algorítmicas sobre Arquivo

Exemplo de algoritmo que cria e armazena registro em um arquivo direto.

```
algoritmo "Gravar dados em arquivo direto"
  declare registro PRODUTOS
  {
    inteiro código chave;
    literal nome;
  }
  define arquivo LOJA direto de PRODUTOS;
  declare variável códigoAuxiliar;

  abrir LOJA;           // abre arquivo em modo leitura e escrita (ambos)
  enquanto (novoRegistro = "sim") faça
    escreva ("Informe o código do Produto?");
    leia (códigoAuxiliar);
    enquanto (não EOF) faça    // verifica se não é fim de arquivo
      se (recuperar LOJA.PRODUTOS.código = códigoAuxiliar) então
        escreva ("Código do Produto já cadastrado.");
```

# Operações Algorítmicas sobre Arquivo

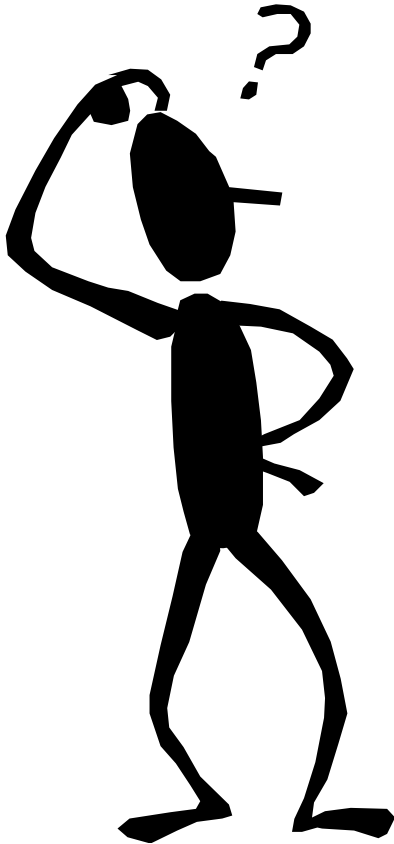
// continuação do algoritmo de gravação no arquivo direto

```
    interrompe processo;
    fimSe;
    fimEnquanto;
    se (EOF) então           // verifica se chegou no fim do arquivo
        leia PRODUTOS.nome;   // le somente o campo nome de 1 registro
        // grava abaixo 1 registro completo de Produto no arquivo
        inserir LOJA.PRODUTOS(códigoAuxiliar, nome);
    fimSe;
    fimEnquanto;
    fechar LOJA;
    fimAlgoritmo
```

- Arquivo direto possui uma estrutura auxiliar que guarda a chave associada a posição do registro no arquivo de dados



# Exercício Proposto



- 1) Faça um programa na Linguagem C que manipulará um arquivo SEQUENCIAL de dados usando estruturas heterogêneas (*struct*) coerentes ao armazenamento de registros lógicos relacionados a situação que envolvem pessoas proprietárias de automóveis (só Carros).

Sua solução deverá guardar estes dados de maneira eficiente a recuperação e novos cadastros no(s) arquivo(s) sequencial(is) de dados.

# Referência de Criação e Apoio ao Estudo

## Material para Consulta e Apoio ao Conteúdo

- SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. Sistemas de Banco de Dados, 3ª ed. 1999.
  - Capítulo 10
- Universidade de Brasília (UnB Gama)
  - <https://sae.unb.br/cae/conteudo/unbfga/>  
(escolha a disciplina **Algoritmo Prog Computadores** e a opção **Arquivos** no menu lateral à esquerda)
- Universidade de São Paulo (USP - ICMC)
  - <http://wiki.icmc.usp.br/images/0/0d/SCC0215012015camposRegistros.pdf>
- PORTARIA JÚNIOR, S.C. Apostila de Banco de Dados
  - <http://www.sergioportari.com.br/wp-content/uploads/2016/07/Aula03-Cap.-2-Organizacao-de-Arquivos.pdf>