

Algoritmos e Estruturas de Dados III

Aula Aula 3.1 - Arquivos Sequenciais

Prof. Felipe Lara



Roteiro do Conteúdo



3.1 Arquivos

- Arquivos Sequenciais
- Chaves de Ordenação
- Chave Primária
- Operações em Arquivos (CRUD)

3.2 Ordenação de Arquivos

- Ordenação Externa
 - Intercalação Balanceada
 - Intercalação Polifásica

Roteiro do Conteúdo



3.1 Arquivos

- Arquivos Sequenciais
- Chaves de Ordenação
- Chave Primária
- Operações em Arquivos (CRUD)

3.2 Ordenação de Arquivos

- Ordenação Externa
 - Intercalação Balanceada
 - Intercalação Polifásica

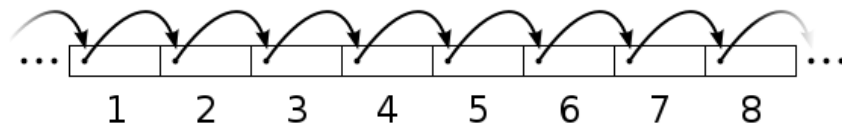
Roteiro da Aula

- Arquivos
 - Arquivos Sequenciais
 - Chaves de Ordenação
 - Chave Primária
 - Operações em Arquivos (CRUD)

Arquivos Sequenciais

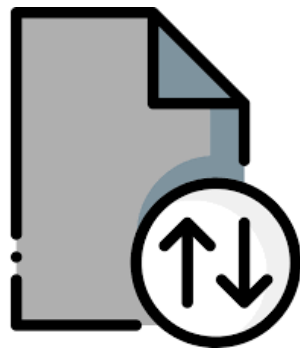
- Arquivos em que os registros são **acessados na ordem em que estão armazenados**.
 - Normalmente são usados quando há pouca (ou nenhuma) movimentação de registros.
- O objetivo é o acesso rápido a um conjunto de registros.
 - Não são bons para acessos aleatórios

Sequential access



Arquivos Sequenciais - Ordenação

- Geralmente existe algum critério na organização dos registros, isto é, algo que define a ordem deles:
 - Momento (data/hora) de criação
 - Chave primária (atributo ou conjunto de atributos que identifica a entidade de forma exclusiva)
 - Qualquer outra **chave de ordenação**
 - Nenhuma ordem



Arquivos Sequenciais – Chave de Ordenação

- A chave de ordenação é o atributo ou conjunto de atributos que estabelece a ordem dos registros
- Tipos de chave
 - Atributo (ex.: ID, CPF, CNPJ, RG, ...)
 - Combinação de campos (ex.: estado+cidade)
 - Processamento de campos (ex.: inversão de data)

Arquivos Sequenciais – Exemplo

ID	Nome	Email	Data Nascimento
1	Carlos Chagas	chagas@gmail.com	13/12/1977
2	Ana Maria Fonseca	ana@hotmail.com	17/04/1975
3	Sílvia Costa	silvia@gmail.com	28/01/1969
4	Joana Torres	joana@gmail.com	31/07/1982

Arquivos Sequenciais – Exemplo

Ordenação por ID



ID	Nome	Email	Data Nascimento
1	Carlos Chagas	chagas@gmail.com	13/12/1977
2	Ana Maria Fonseca	ana@hotmail.com	17/04/1975
3	Sílvia Costa	silvia@gmail.com	28/01/1969
4	Joana Torres	joana@gmail.com	31/07/1982

Arquivos Sequenciais – Exemplo

Ordenação por Nome



ID	Nome	Email	Data Nascimento
2	Ana Maria Fonseca	ana@hotmail.com	17/04/1975
1	Carlos Chagas	chagas@gmail.com	13/12/1977
4	Joana Torres	joana@gmail.com	31/07/1982
3	Sílvio Costa	silvio@gmail.com	28/01/1969

Arquivos Sequenciais – Exemplo

Ordenação por
Sobrenome



ID	Nome	Email	Data Nascimento
1	Carlos Chagas	chagas@gmail.com	13/12/1977
3	Sílvio Costa	silvio@gmail.com	28/01/1969
2	Ana Maria Fonseca	ana@hotmail.com	17/04/1975
4	Joana Torres	joana@gmail.com	31/07/1982

**Chave de
Ordenação**

Chagas

Costa

Fonseca

Torres

Arquivos Sequenciais – Exemplo

Ordenação por data
nasc. decrescente



ID	Nome	Email	Data Nascimento
4	Joana Torres	joana@gmail.com	31/07/1982
1	Carlos Chagas	chagas@gmail.com	13/12/1977
2	Ana Maria Fonseca	ana@hotmail.com	17/04/1975
3	Sílvio Costa	silvio@gmail.com	28/01/1969

**Chave de
Ordenação**

1982 07 31

1977 12 13

1975 04 17

1969 01 28

Arquivos Sequenciais - Observações

- O arquivo eventualmente precisará ser reordenado para se manter a ordem dos registros
 - Arquivos sequenciais são usados quando sofrem poucas alterações

Arquivos Sequenciais - Observações

- O arquivo eventualmente precisará ser reordenado para se manter a ordem dos registros
 - Arquivos sequenciais são usados quando sofrem poucas alterações
- Arquivos sequenciais geralmente são usados como arquivos temporários
 - Existem outras estruturas bem melhores para **acesso aleatório** como os arquivos indexados e os arquivos diretos

Arquivos Sequenciais – Chaves Primárias

Como escolher a chave primária?

- Chaves candidatas
 - Chaves que identificam cada entidade de forma exclusiva
 - Ex.: CPF, CNPJ, RG, Matrícula, E-Mail, Telefone, etc.
- Chave primária
 - Chave candidata escolhida

Arquivos Sequenciais – Chaves Primárias

Problemas das chaves candidatas

- Nome: Algumas pessoas têm o mesmo nome e outras mudam de nome quando se casam
- E-mail: As pessoas trocam seus e-mails
- Telefone: São reutilizáveis
- CPF e CNPJ: São muito longos

Arquivos Sequenciais – Chaves Primárias

Características Identificadores (ID):

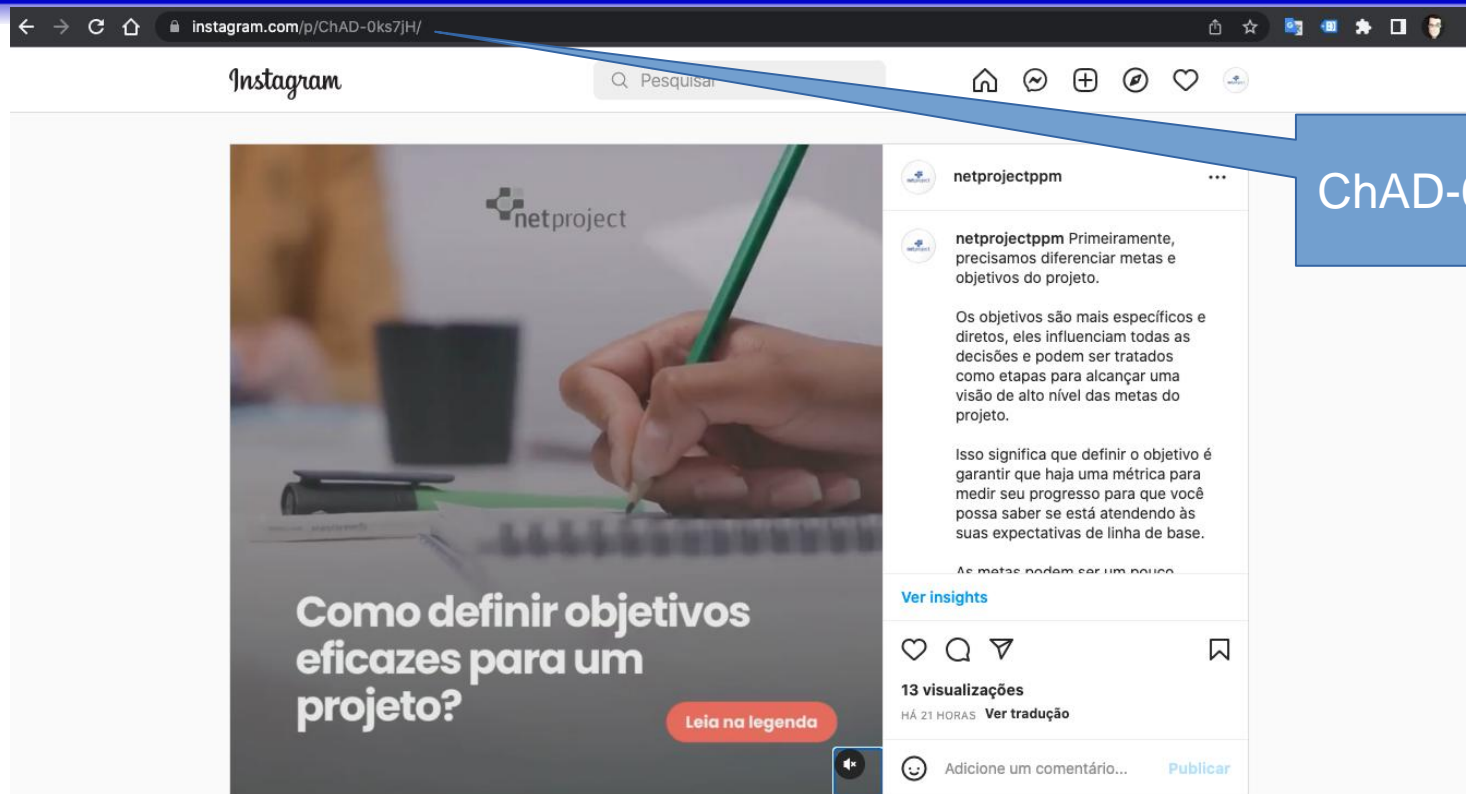
- Numéricos (sem sinal) melhor aproveitamento dos bytes
- Sequenciais evita o desperdício de valores numéricos
- Exclusivos não podem ser ambíguos
- Não significativos não podem ser alterados
- Não reusáveis nunca são reaproveitados

Arquivos Sequenciais – Chaves Primárias


Características Identificadores (ID):

- Numéricos (sem sinal) melhor aproveitamento dos bytes
 - Sequenciais evita o desperdício de valores numéricos
 - Exclusivos não podem ser ambíguos
 - Não significativos não podem ser alterados
 - Não reusáveis nunca são reaproveitados
-
- Os identificadores são chaves internas, isto é, só são usadas pelo sistema e não devem ser usados pelos usuários
 - Os usuários continuarão buscando as entidades por meio de e-mails, CPFs, CNPJs, etc.

Arquivos Sequenciais – Identificadores



Arquivos Sequenciais – Identificadores

Olá
Selecione o endereço

Todos

Olá,
Cor

Todos Venda na Amazon Mais Vendidos Ofertas do Dia Prime Atendimento ao Cliente Livros Música Novidades na Amazon Eletrônicos

1-48 de 787 resultados para "nintendo switch"

Elegível a Frete Grátis

☐ Frete Grátis em envios pela Amazon

Todos os clientes têm frete GRÁTIS em pedidos acima de R\$ 99 para livros, e R\$149 em outros itens enviados pela Amazon

Departamento

Games e Consoles

Consoles Nintendo Switch

Acessórios para Nintendo Switch

Jogos para Nintendo Switch

Ver todos os 9 Departamentos

Avaliação do Cliente

★★★★★ e acima

★★★★★ e acima

★★★★★ e acima

★★★★★ e acima

Marca

☐ Nintendo

☐ HEYSTOP


☐ HORI

☐ NC Games

☐ SanDisk

☐ PowerA

RESULTADOS



Console New Nintendo Switch - Azul Neon e Vermelho Neon (versão nacional)

19 set 2020


★★★★★ ~ 1.748

Nintendo Switch

-19% R\$ 2.349⁹⁰ R\$2.899,00

em até 10x de R\$ 234,99 sem juros

Receba Sexta-feira, 18 de set



Console Nintendo Switch OLED - Branco por Nintendo


★★★★★ ~ 15.255

Nintendo Switch

R\$ 2.682²⁶

em até 10x de R\$ 268,28 sem juros

Receba Sexta-feira, 26 de ago - Sexta-feira, 2 de set



Console Nintendo Switch - Azul Neon e Vermelho Neon (Nacional)

1 out 2020

★★★★★ ~ 2.660

Nintendo Switch

R\$ 2.399⁰⁰

em até 10x de R\$ 239,90 sem juros

prime Receba até Quinta-feira, 18 de ago

Arquivos Sequenciais – Identificadores

https://www.amazon.com.br/Console-New-Nintendo-Switch-Vermelho/dp/B08M9S4XTR/ref=sr_1_1?crd=1LBQZAFP5U9CU&keywords=nintendo+switch&qid=1660065826&sprefix

Olá, Faça seu perfil
Selecione o endereço

Todos nintendo switch

Amazon Music UNLIMITED

Mais Vendidos Ofertas do Dia Prime Atendimento ao Cliente Livros Música Novidades na Amazon Eletrônicos


PlayStation 4 Xbox One Nintendo Switch Acessórios PC SteelBooks

HEYSTOP Estojo de transporte para Nintendo Switch com protetor de tela, kit de acessórios...
Cupom de 5% off
★★★★★ 28
R\$119,00

Console New Nintendo Switch - Neon e Vermelho Neon (versão nacional)
Marca: Nintendo
Plataforma : Nintendo Switch
★★★★★ 1.748 avaliações de clientes

-19% R\$2.349⁹⁰
De: R\$2.899,00

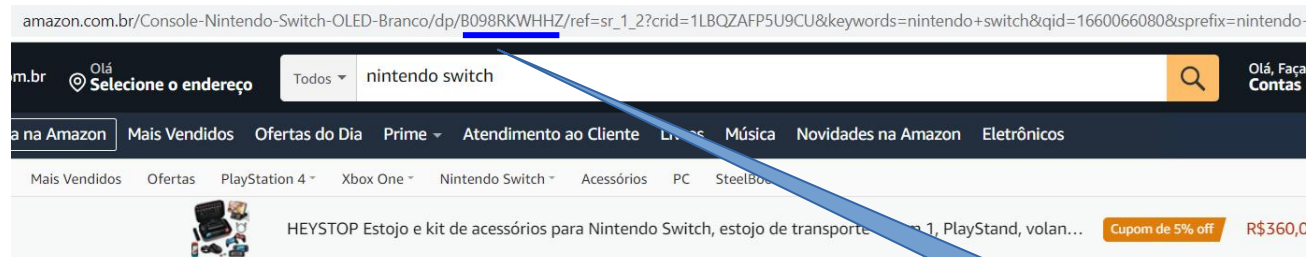
Pagamentos e Segurança Política de devolução



A blue arrow points from the product identifier 'B08M9S4XTR' in a box to the product's ASIN in the URL and the product title.

B08M9S4XTR

Arquivos Sequenciais – Identificadores



Console Nintendo Switch OLED - Branco

Marca: Nintendo

Plataforma: Nintendo Switch

★★★★★ 15.255 avaliações de clientes

R\$ 2.682,26



Pagamentos e
Segurança



Política de
devolução

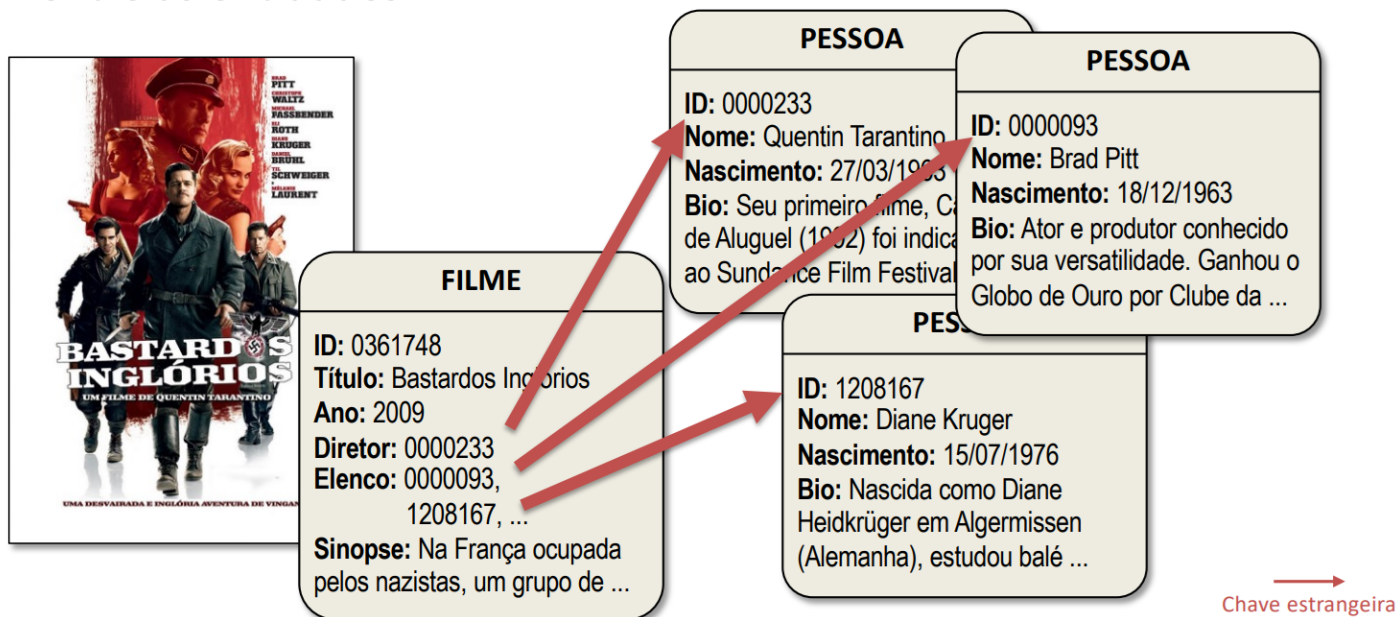
Sobre este item

- Console Nintendo Switch modelo OLED: Joy-Cons e Dock na cor branca;
- Tela OLED de 7";
- Memória interna de 64GB;

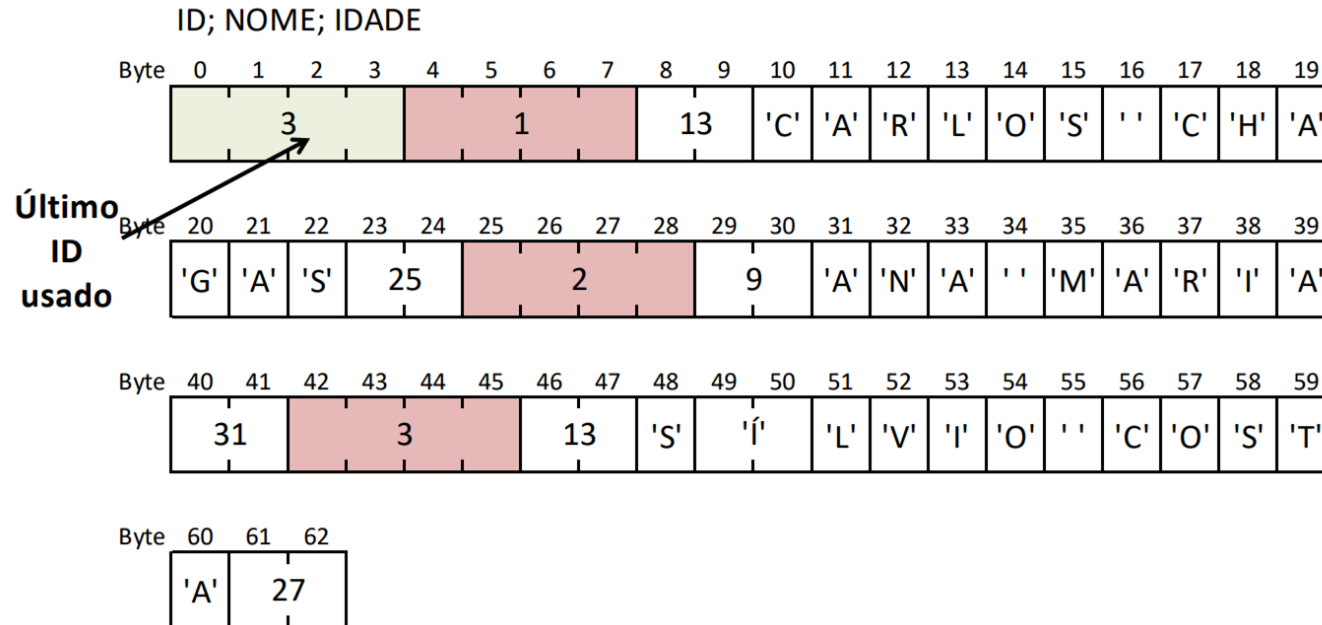
B098RKWHH7

Arquivos Sequenciais – Identificadores

Os identificadores ajudam a estabelecer os **relacionamentos** entre as entidades



Arquivos Sequenciais - Cabeçalho do arquivo



Operações em arquivos sequenciais



CREATE



READ



UPDATE



DELETE

C

R

U

D



PUC Minas

CRUD

Operação	SQL	HTTP
Create	INSERT	POST / PUT
Read	SELECT	GET
Update	UPDATE	PUT / PATCH
Delete	DELETE	DELETE

Interface do CRUD

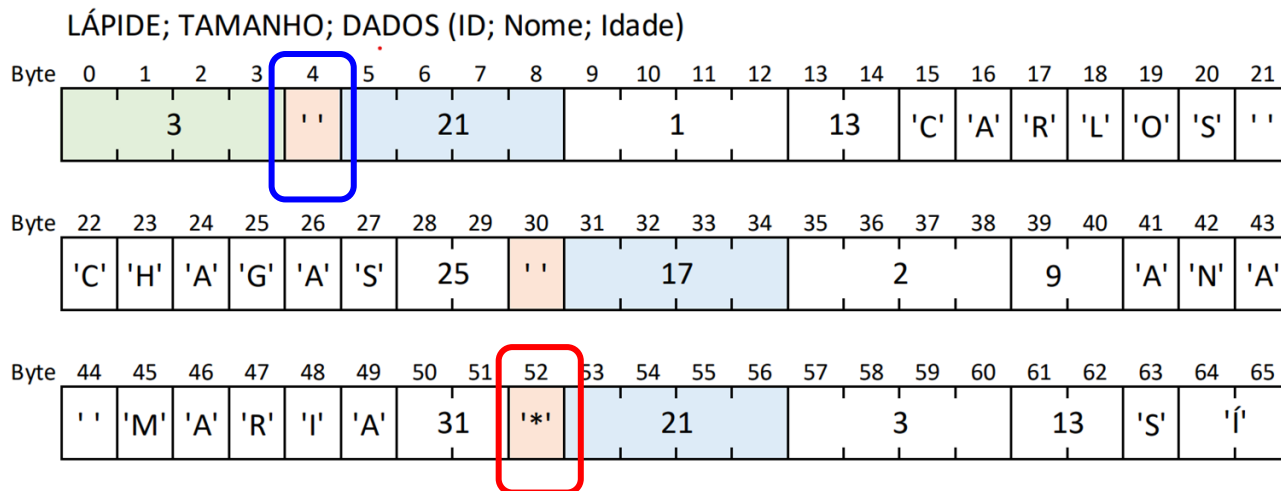
O ID passado nesse novo objeto será ignorado (usar -1)

- $ID \leftarrow \text{arquivo.create}(\text{novo_objeto})$
- $\text{objeto} \leftarrow \text{arquivo.read}(ID)$
- $ok \leftarrow \text{arquivo.update}(\text{objeto_atualizado})$
- $ok \leftarrow \text{arquivo.delete}(ID)$

O ID deve permanecer o mesmo, pois IDs nunca são alterados.

CRUD

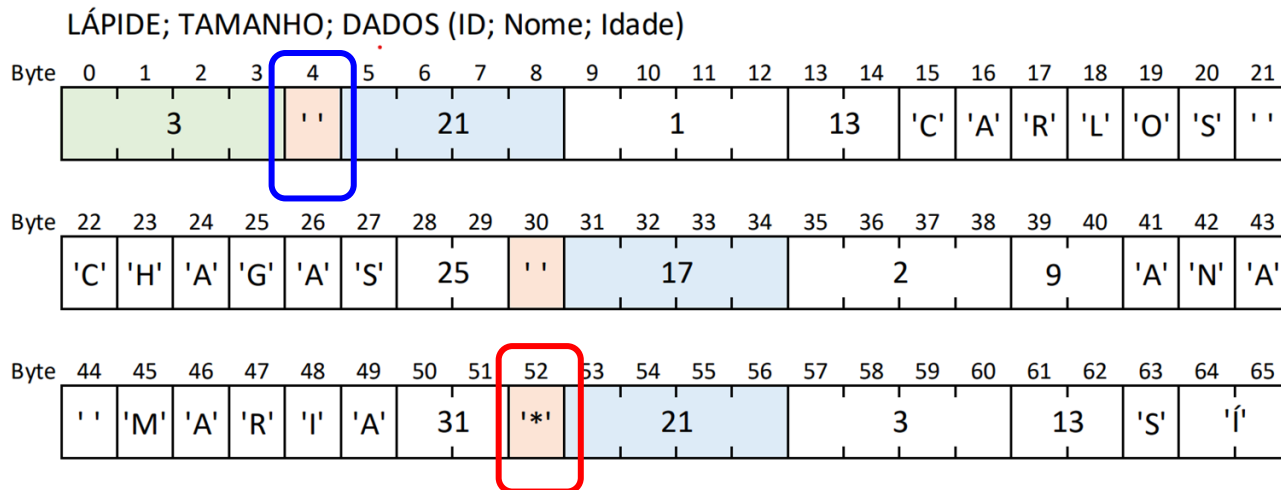
- A operação 'excluir' (delete) não é física, mas lógica.
- Lápide
 - Marca de exclusão
 - Campo (1 byte) que indica se o registro foi excluído ou permanece válido
 - Os registros só são realmente apagados do arquivo quando há uma reordenação



CRUD

- A operação 'excluir' (delete) não é física, mas lógica.
- Lápide
 - Marca de exclusão
 - Campo (1 byte) que indica se o registro foi excluído ou permanece válido
 - Os registros só são realmente apagados do arquivo quando há uma reordenação

Obs: O espaço do disco aparentemente excluído fica livre para armazenar novos dados



CRUD

LÁPIDE; TAMANHO; DADOS (ID; Nome; Idade)

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
			3		' '			21				1			13	'C'	'A'	'R'	'L'	'O'	'S'	' '

Byte	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43
	'C'	'H'	'A'	'G'	'A'	'S'	25		' '			17			2			9		'A'	'N'	'A'

Byte	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65
	' '	'M'	'A'	'R'	'I'	'A'	31		'*'			21			3			13		'S'		'Í'

Byte	66	67	68	69	70	71	72	73	74	75	76	77
	'L'	'V'	'I'	'O'	' '	'C'	'O'	'S'	'T'	'A'		27

CRUD - Algoritmo Create (inclusão)

```
01: algoritmo create(objeto)
02:   |   mover o ponteiro para início do arquivo (cabeçalho)
03:   |   ler últimoID
04:   |   objeto.ID  $\leftarrow$  últimoID + 1
05:   |   mover o ponteiro para início do arquivo
06:   |   escrever objeto.ID
07:   |   criar registro para o objeto
08:   |   mover para o fim do arquivo
09:   |   escrever registro
10: fim-algoritmo
```

CRUD - Algoritmo Read (busca sequencial)

```
01: algoritmo read(ID)
02:   mover o ponteiro para o primeiro registro (após o cabeçalho)
03:   enquanto não atingir o fim do arquivo
04:     ler próximo registro
05:     se registro.lapide ≠ '*'
06:       então extrair objeto do registro
07:       se objeto.ID = ID
08:         então retornar objeto e terminar
09:       fim-se
10:     fim-se
11:   fim-enquanto
12:   retornar objeto vazio // null
13: fim-algoritmo
```


CRUD - Algoritmo Read (busca sequencial)

Obs: leitura de um conjunto de objetos

```
01: algoritmo read(critérios)
02:   criar conjunto vazio
03:   mover o ponteiro para o primeiro registro (após o cabeçalho)
04:   enquanto não atingir o fim do arquivo
05:     ler próximo registro
06:     se registro.lapide ≠ '*'
07:       então extrair objeto do registro
08:       se registro atender aos critérios
09:         | então adicionar objeto ao conjunto
10:       fim-se
11:     fim-se
12:   fim-enquanto
13:   retornar conjunto
14: fim-algoritmo
```

CRUD - Algoritmo Delete (exclusão)

```
01: algoritmo delete(ID)
02:   mover o ponteiro para o primeiro registro (após o cabeçalho)
03:   enquanto não atingir o fim do arquivo
04:     pos ← posição do ponteiro
05:     ler próximo registro
06:     se registro.lapide ≠ '*'
07:       então extrair objeto do registro
08:       se objeto.ID = ID
09:         então mover para pos
10:         escrever lápide como excluído
11:         retornar verdadeiro e terminar
12:       fim-se
13:     fim-se
14:   fim-enquanto
15:   retornar falso
16: fim-algoritmo
```

CRUD - Algoritmo Update (atualização)

Se a alteração não implicar em mudança de tamanho

- Escrever o registro alterado na mesma posição

Se o registro diminuir de tamanho

- Escrever o registro na mesma posição, mas mantendo o mesmo indicador de tamanho

Se o registro aumentar de tamanho

- Marcar o registro atual como excluído e criar um novo registro no fim do arquivo

```
01: algoritmo update(novoObjeto)
02:   mover para o primeiro registro do arquivo (após cabeçalho)
03:   enquanto não atingir o fim do arquivo
04:     pos ← posição do ponteiro
05:     ler próximo registro
06:     se registro.lapide ≠ '*'
07:       então extrair objeto do registro
08:       se objeto.ID = novoObjeto.ID
09:         então criar novoRegistro para novoObjeto
10:           se novoRegistro.tamanho ≤ registro.tamanho
11:             então mover para pos
12:               escrever novoRegistro mantendo ind.tam.
13:             senão mover para pos
14:               escrever lápide como excluído
15:               mover para fim do arquivo
16:               escrever novoRegistro
17:             fim-se
18:           retornar verdadeiro e terminar
19:         fim-se
20:       fim-se
21:     fim-enquanto
22:     retornar falso
23: fim-algoritmo
```

CRUD - Considerações finais

- Arquivos sequenciais dependem de **acesso sequencial**, o que significa que eles não são bons para quaisquer operações de acesso aleatório

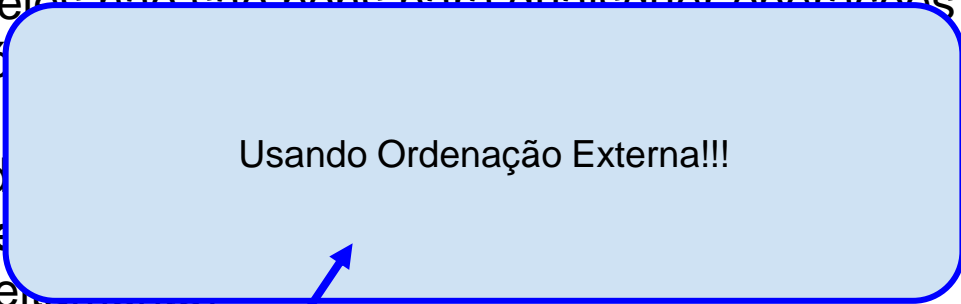
CRUD - Considerações finais

- Arquivos sequenciais dependem de **acesso sequencial**, o que significa que eles não são bons para quaisquer operações de acesso aleatório
- Os espaços deixados pelos registros **excluídos são espaços que podem ser reaproveitados** (desde que exista uma lógica para esse reaproveitamento)

CRUD - Considerações finais

- Arquivos sequenciais dependem de **acesso sequencial**, o que significa que eles não são bons para quaisquer operações de acesso aleatório
- Os espaços deixados pelos registros **excluídos são espaços que podem ser reaproveitados** (desde que exista uma lógica para esse reaproveitamento)
- O arquivo deve ser **reordenado** sempre que necessário (ordenação externa)

CRUD - Considerações finais

- Arquivos sequenciais dependem de **acesso sequencial**, o que significa que eles não são bons para quaisquer operações de acesso aleatório
- Os espaços de armazenamento podem ser reutilizados para esse reaproveitamento,  Usando Ordenação Externa!!!
- O arquivo deve ser **reordenado** sempre que necessário (ordenação externa)