

# SBVORIN: Organização e Recuperação da Informação

## Aula 04: Arquivos e Ordenações Externas

Bacharelado em Ciência da Computação  
Prof. Dr. David Buzatto



INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
SÃO PAULO  
Campus São João da Boa Vista

# Arquivos

- Os arquivos são normalmente armazenados em algum meio durável de armazenamento:
  - Fitras magnéticas, discos magnéticos flexíveis (disquetes), discos magnéticos rígidos (HD), drives de estado sólido (SSD), cartões de memória, pen drives, mídias óticas (CD, DVD, Blu-Ray etc.) etc.
- Dependendo da aplicação computacional empregada na leitura e escrita de um arquivo, há um formato associado ao mesmo, mas basicamente os arquivos são de texto ou binários;
- Dependendo da linguagem usada na manipulação de arquivos, pode haver suporte nativo mais ou menos sofisticado/abstrato ou existirem bibliotecas mais específicas do que o suporte fornecido pela API da linguagem de programação em questão.

# Arquivos

- ▶ Em Java, de forma nativa, a manipulação se dá através das classes essencialmente contidas nos pacotes:
  - ▶ **java.io**: manipulação de fluxos de dados de entrada e saída do sistema, além de mecanismos de serialização/desserialização e comunicação com o sistema de arquivos do sistema operacional;
    - ▶ [Referência](#)
  - ▶ **java.nio** (e vários subpacotes): definição de *buffers* (contêineres de dados), *charsets*, canais e seletores para E/S não bloqueante (*non-blocking I/O*);
    - ▶ [Referência](#)
- ▶ No repositório de estruturas de dados e algoritmos há exemplos de manipulação de arquivos de texto, binário, de acesso randômico e serialização no pacote **aesd.algorithms.files**, que nos dará base para entender o algoritmo de ordenação externa que será estudado.

# Ordenações Externas

- Consistem em ordenar arquivos de tamanho maior que a memória interna disponível;
- Os algoritmos devem diminuir o número de acesso às unidades de memória externa;
- Os dados dos arquivos são armazenados de forma sequencial;
- Apenas um registro armazenado pode ser acessado em um dado momento;
- Em comparação com as ordenações:
  - O custo para acessar os itens que serão ordenados é algumas ordens de grandeza maior;
  - O custo principal é relacionado à transferência de dados entre a memória interna e a externa.

# Ordenações Externas

- A Ordenação por Intercalação é o método mais importante;
  - Intercalar: combinar dois ou mais blocos ordenados em um único bloco ordenado (lembre-se do Merge Sort);
  - A intercalação é usada como uma operação auxiliar;
- Estratégia geral:
  - Quebrar o arquivo em blocos do tamanho da memória interna disponível;
  - Ordenar cada bloco de memória;
  - Intercalar os blocos ordenados, realizando diversas passadas pelo arquivo;
- Importante:
  - Reduzir o número de passadas sobre o arquivo, usando mais memória interna;
  - O número de leituras e escritas em memória auxiliar é uma boa medida de complexidade;
  - Bons algoritmos de ordenação externa envolvem menos de dez passadas pelo arquivo.



# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

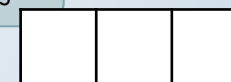
Arquivo com 22 registros

I N T E R C A L A C A O B A L A N C E A D A

6 unidades de fita magnética disponíveis

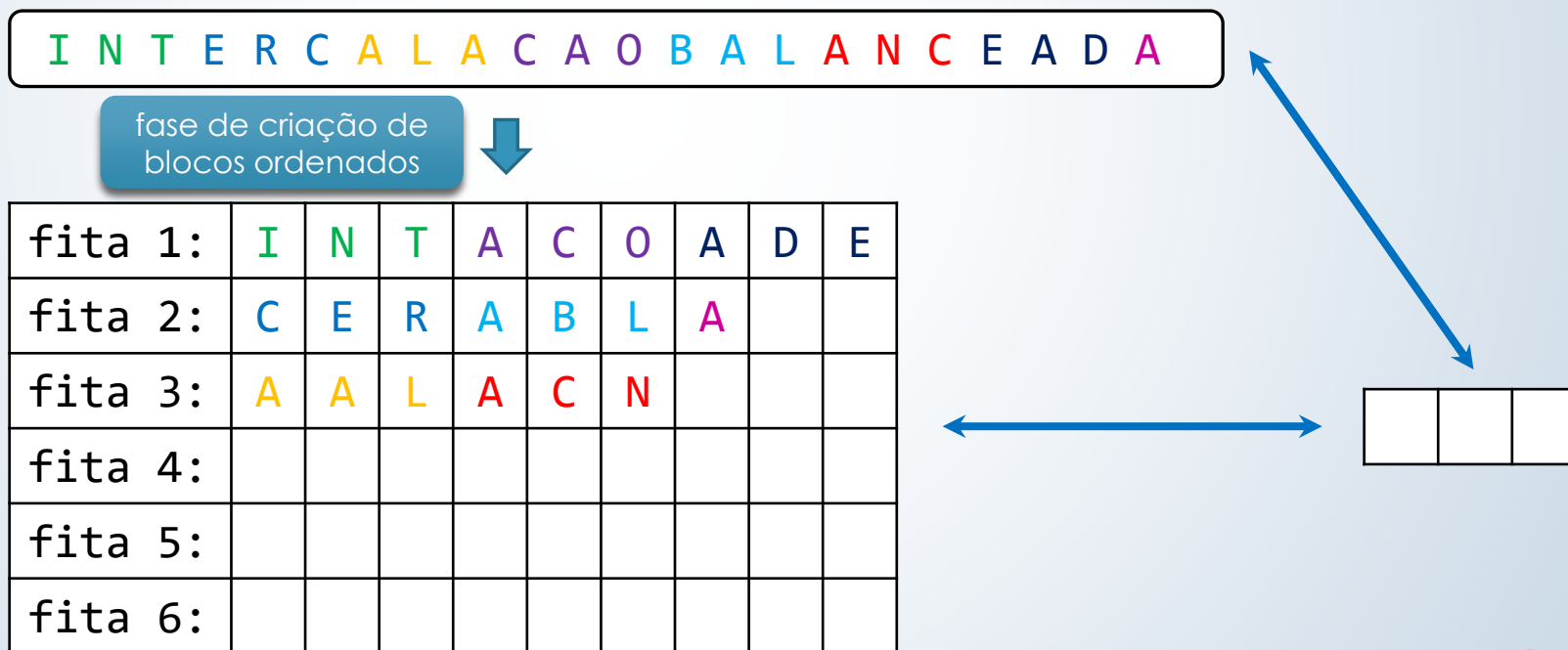
fita 1:									
fita 2:									
fita 3:									
fita 4:									
fita 5:									
fita 6:									

Memória interna  
com capacidade  
para 3 registros



# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída



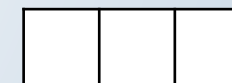
# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

I N T E R C A L A C A O B A L A N C E A D A

fase de  
intercalação

fita 1:	I	N	T	A	C	O	A	D	E
fita 2:	C	E	R	A	B	L	A		
fita 3:	A	A	L	A	C	N			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:									
fita 6:									





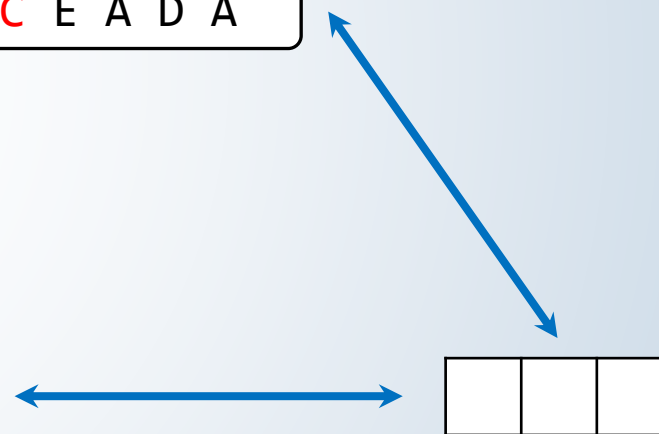
# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

I N T E R C A L A C A O B A L A N C E A D A

fase de  
intercalação

fita 1:	I	N	T	A	C	O	A	D	E
fita 2:	C	E	R	A	B	L	A		
fita 3:	A	A	L	A	C	N			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:									



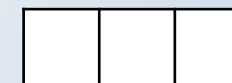
# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

I N T E R C A L A C A O B A L A N C E A D A

fase de  
intercalação

fita 1:	I	N	T	A	C	O	A	D	E
fita 2:	C	E	R	A	B	L	A		
fita 3:	A	A	L	A	C	N			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



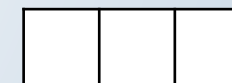
# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

I N T E R C A L A C A O B A L A N C E A D A

fita 1:	I	N	T	A	C	O	A	D	E
fita 2:	C	E	R	A	B	L	A		
fita 3:	A	A	L	A	C	N			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					

resultado da  
primeira passada  
da segunda fase



# Ordenações Externas

- ▶ Quantas passadas são necessárias para ordenar um arquivo de tamanho arbitrário?
- ▶ Seja:
  - ▶  $n$ : quantidade de registros do arquivo;
  - ▶  $m$ : quantidade de palavras na memória interna ocupadas pelos registros;
  - ▶  $\frac{n}{m}$ : quantidade de blocos ordenados produzidos na primeira etapa;
  - ▶  $P(n)$ : o número de passadas para a fase de intercalação;
  - ▶  $f$ : o número de fitas usadas em cada passada;
- ▶ Assim temos:  $P(n) = \log_f \frac{n}{m}$
- ▶ Para o exemplo,  $n = 22$ ,  $m = 3$  e  $f = 3$ , temos:  $P(n) = \log_3 \frac{22}{3} = 2$

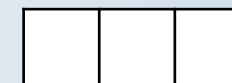
# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

I N T E R C A L A C A O B A L A N C E A D A

fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	C	E	R	A	B	L	A		
fita 3:	A	A	L	A	C	N			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					

fase de intercalação  
(segunda passada)



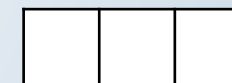
# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

I N T E R C A L A C A O B A L A N C E A D A

fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	A	A	L	A	C	N			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					

fase de intercalação  
(segunda passada)





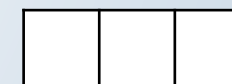
# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

I N T E R C A L A C A O B A L A N C E A D A

fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					

fase de intercalação  
(segunda passada)



# Ordenações Externas

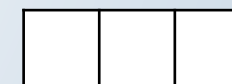
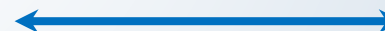
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A N T E R C A L A C A O B A L A N C E A D A

“remontar” o  
arquivo



fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



# Ordenações Externas

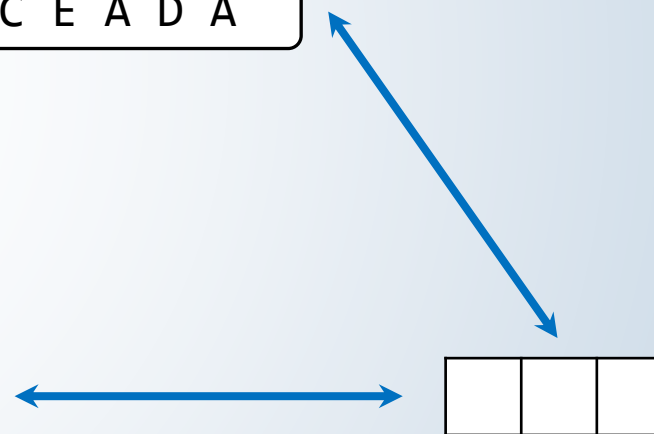
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A T E R C A L A C A O B A L A N C E A D A

“remontar” o  
arquivo



fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



# Ordenações Externas

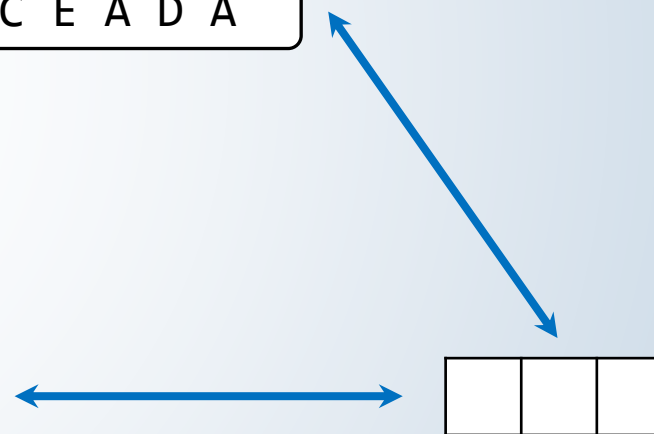
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A A E R C A L A C A O B A L A N C E A D A

“remontar” o  
arquivo



fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



# Ordenações Externas

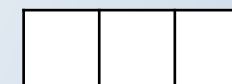
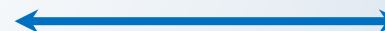
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A A A A A A L A C A O B A L A N C E A D A

“remontar” o  
arquivo



fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



# Ordenações Externas

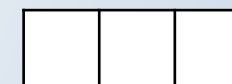
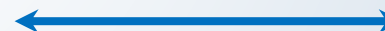
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A A A A A A B A C A O B A L A N C E A D A

“remontar” o  
arquivo



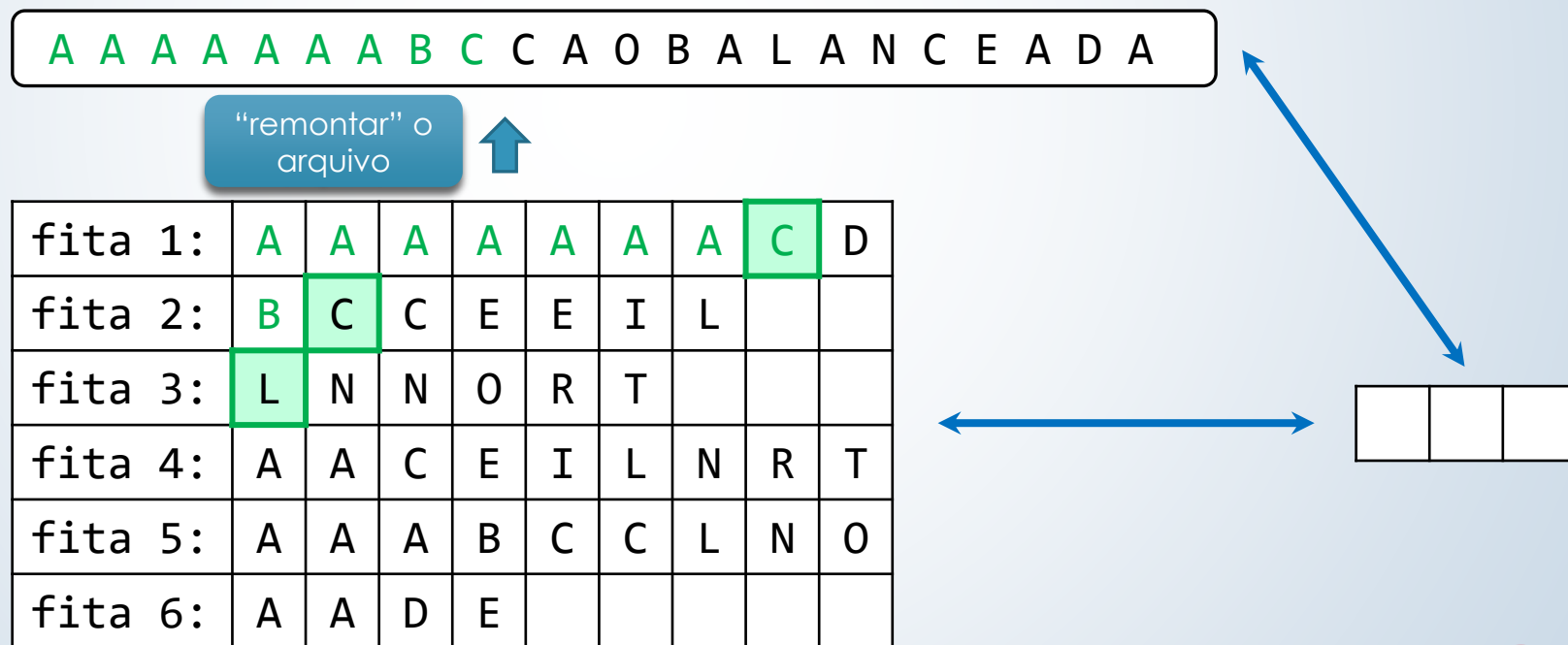
fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					





# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída



# Ordenações Externas

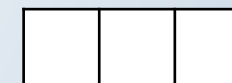
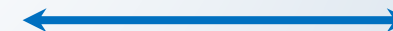
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A A A A A A B C C A O B A L A N C E A D A

“remontar” o  
arquivo



fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



# Ordenações Externas

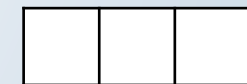
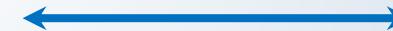
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A A A A A A B C C C O B A L A N C E A D A

“remontar” o  
arquivo



fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



# Ordenações Externas

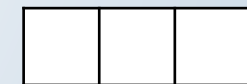
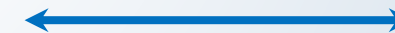
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A A A A A A B C C C D B A L A N C E A D A

“remontar” o  
arquivo



fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



# Ordenações Externas

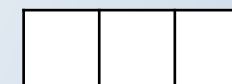
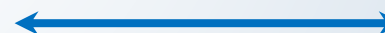
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A A A A A A B C C C D E A L A N C E A D A

“remontar” o  
arquivo



fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



# Ordenações Externas

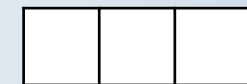
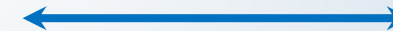
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A A A A A A B C C C D E E L A N C E A D A

“remontar” o  
arquivo



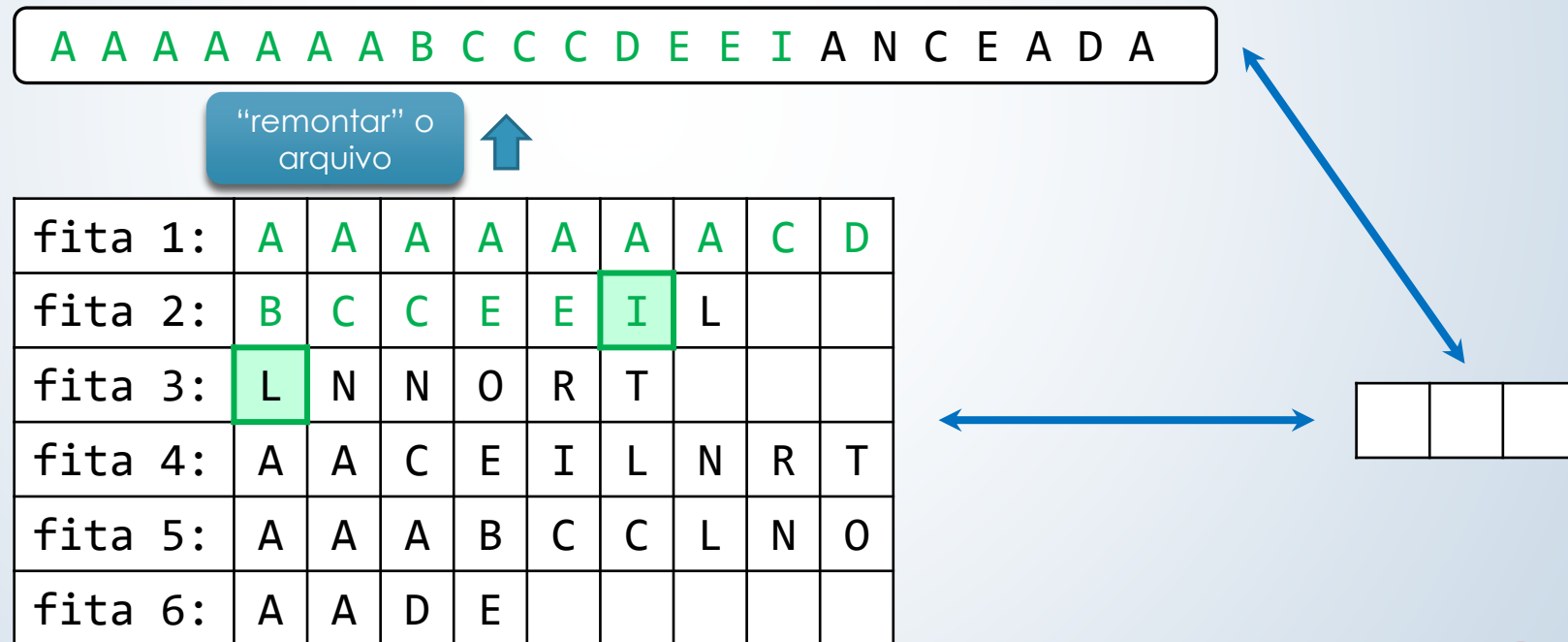
fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					





# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída



# Ordenações Externas

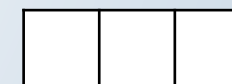
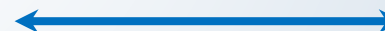
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A A A A A A B C C C D E E I L N C E A D A

“remontar” o  
arquivo

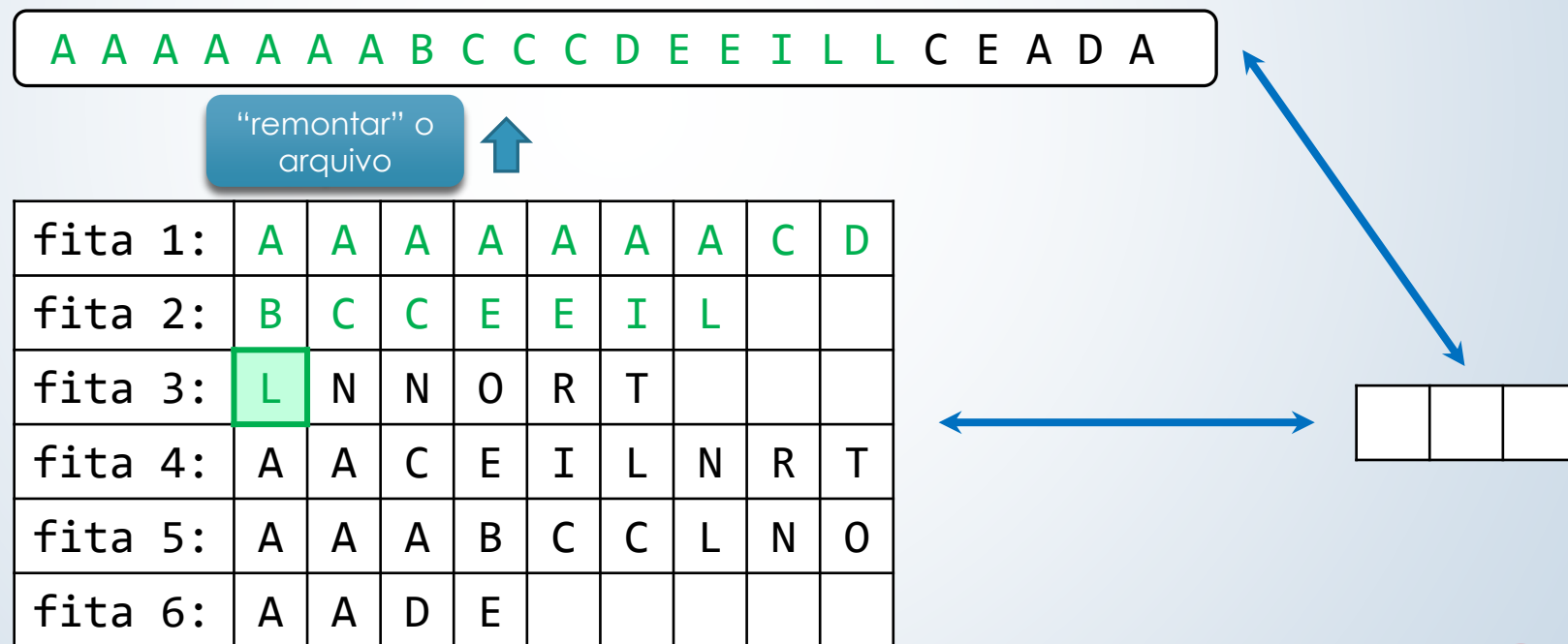


fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída



# Ordenações Externas

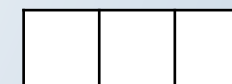
- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída

A A A A A A A B C C C D E E I L L N E A D A

“remontar” o  
arquivo

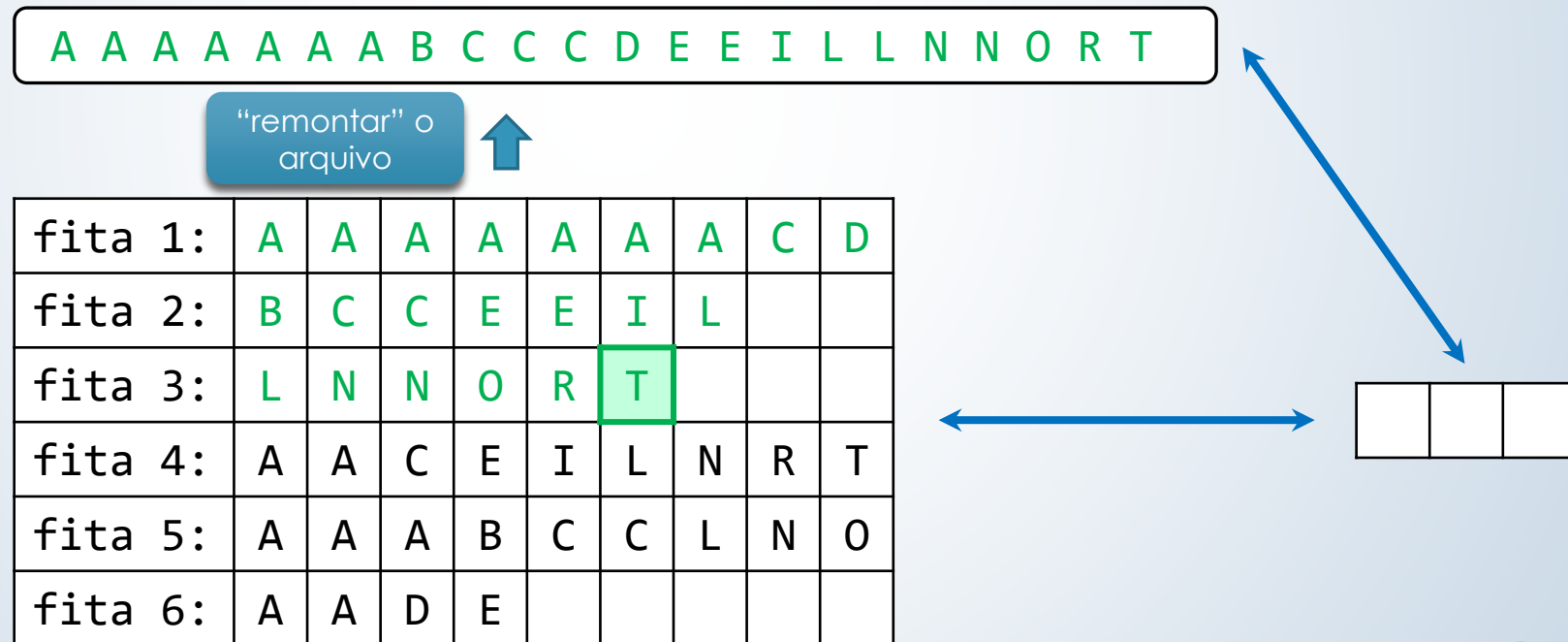


fita 1:	A	A	A	A	A	A	A	C	D
fita 2:	B	C	C	E	E	I	L		
fita 3:	L	N	N	O	R	T			
fita 4:	A	A	C	E	I	L	N	R	T
fita 5:	A	A	A	B	C	C	L	N	O
fita 6:	A	A	D	E					



# Ordenações Externas

- Intercalação Balanceada de Vários Caminhos:
- Exemplo:
  - Objetivo: ordenar 22 registros e coloca-los em uma fita de saída



# Ordenações Externas

- Obviamente, essa abordagem pode ser melhorada usando menos fitas, aplicando outras estruturas de dados como filas de prioridades (seleção por substituição) e aplicando a intercalação polifásica, onde blocos obtidos através de seleção por substituição são intercalados de duas em duas fitas, tornando sempre uma vazia até que a intercalação finalize;
- Não entraremos em mais detalhes, pois partiremos para um exemplo de implementação real do algoritmo Quick Sort Externo, proposto por Monard em 1980 (Maria Carolina Monard, tese de doutorado);
- Caso queira, consulte a obra de Ziviani (2006), apresentada na bibliografia, para mais detalhes em relação à seleção por substituição e à intercalação polifásica.



# Quick Sort Externo

- ▶ Como sua contraparte interna, usa o paradigma de divisão e conquista;
- ▶ Ordena um arquivo  $A = \{R_1, \dots, R_n\}$  de  $n$  registros *in situ*;
- ▶ O armazenamento dos registros é feito de forma consecutiva em memória secundária de acesso randômico, por isso aprendemos a lidar com ponteiros de arquivos em arquivos de acesso randômico no início da aula;
- ▶ O algoritmo usa  $O(\lg n)$  unidades de memória interna, não havendo necessidade de memória externa adicional;
- ▶ Lembre-se que  $\lg n = \log_2 n$ ;

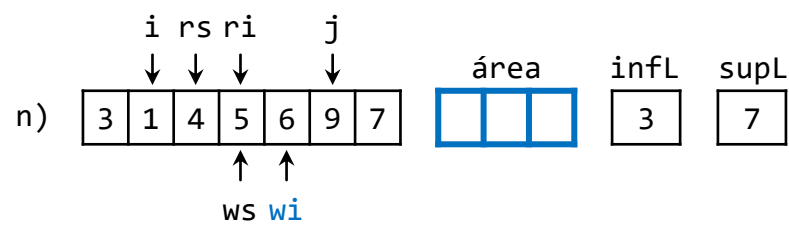
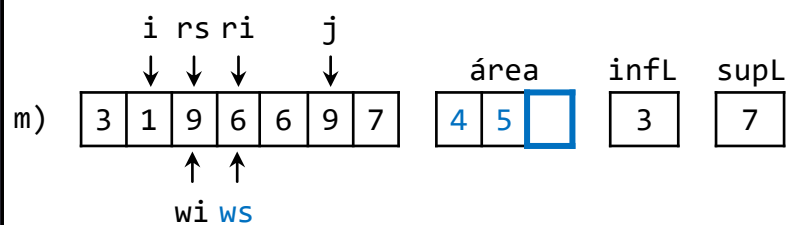
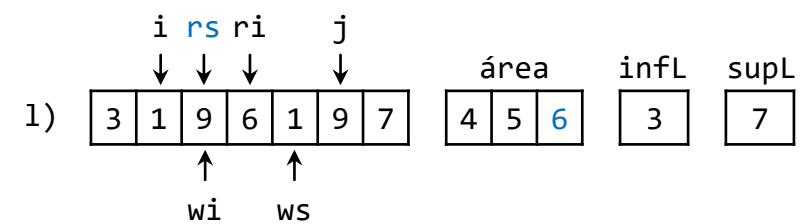
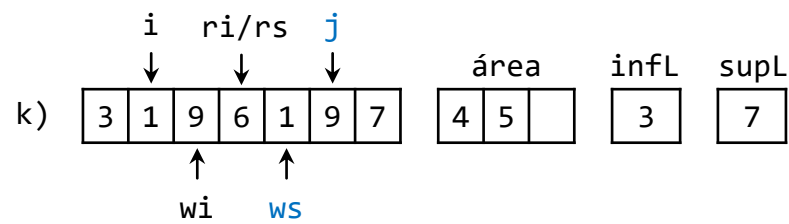
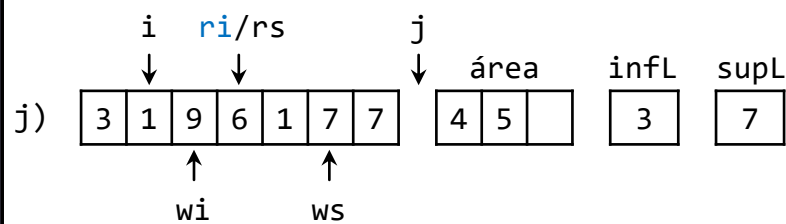
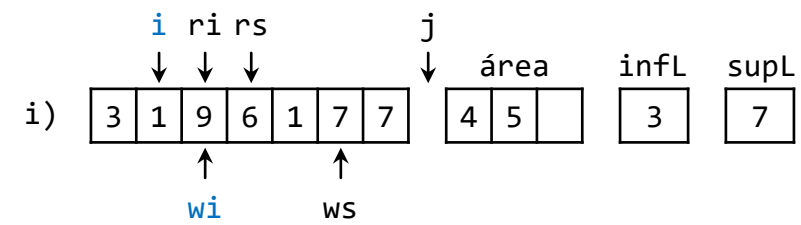
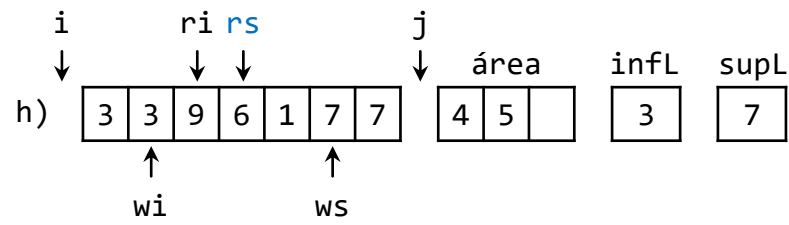
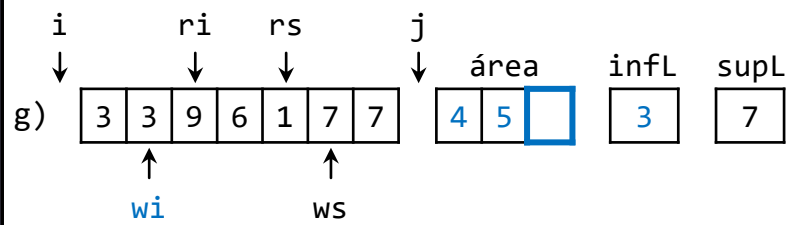
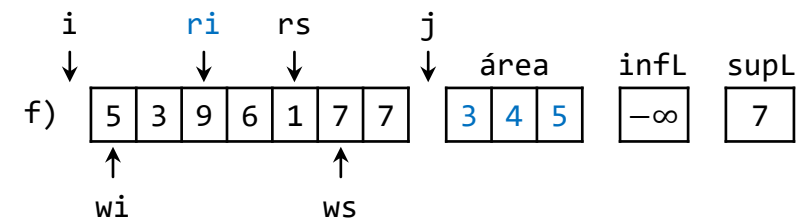
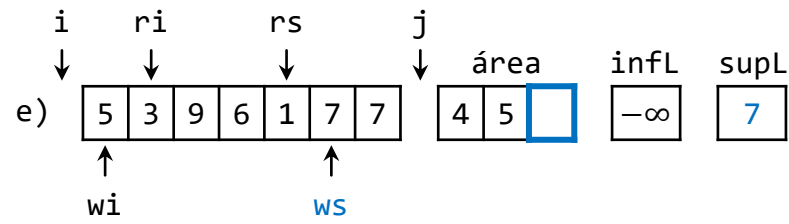
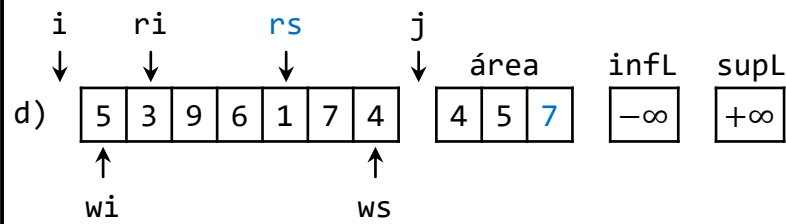
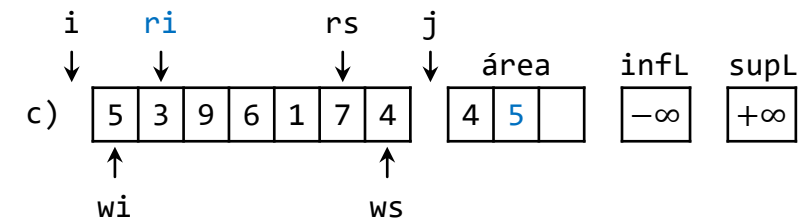
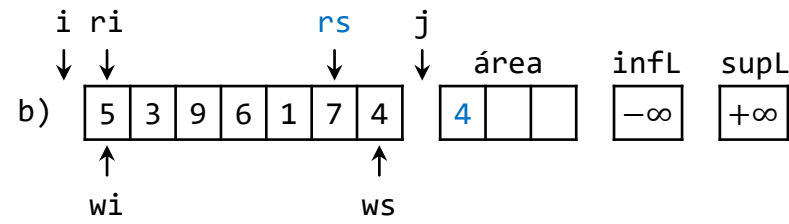
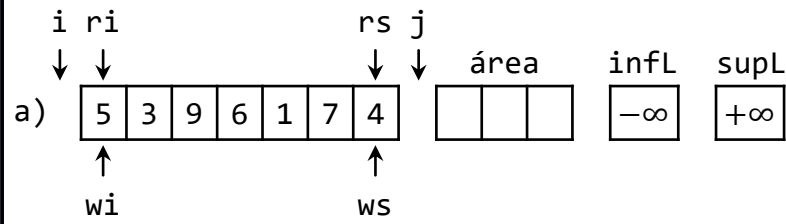
# Quick Sort Externo

- Seja  $R_i$ ,  $1 \leq i \leq n$ , o  $i$ -ésimo registro do arquivo  $A$ ;
- Algoritmo:
  1. Particionar  $A$  da seguinte forma:
    - $\{R_1, \dots, R_i\} \leq R_{i+1} \leq R_{i+2} \leq \dots \leq R_{j-2} \leq R_{j-1} \leq \{R_j, \dots, R_n\}$
  2. Invocar recursivamente o algoritmo em cada um dos subarquivos:
    - $A_1 = \{R_1, \dots, R_i\}$  e  $A_2 = \{R_j, \dots, R_n\}$
- O particionamento usa uma área de armazenamento na memória interna;
  - $T$  é o tamanho da área e é dado por:  $T = j - i - 1$ ,  $T \geq 3$

# Quick Sort Externo

- ▶ Para as chamadas recursivas, consideraremos que:
  - ▶ Primeiro deve-se ordenar os subarquivos de menor tamanho, implicando que, em média,  $O(\lg n)$  subarquivos tenham o processamento postergado;
  - ▶ Subarquivos vazios ou com um único registro são ignorados;
  - ▶ Caso o arquivo de entrada  $A$  possua no máximo  $T$  registros, ele é ordenado em um único passo;
  - ▶ Note a semelhança com o Quick Sort interno.

# Simulação do Quick Sort Externo



Legenda:

ri: read inferior  
rs: read superior  
wi: write inferior  
ws: write superior

ZIVIANI, N. **Projeto de Algoritmos com Implementações em Java e C++**. São Paulo: Cengage, 2006. 644 p.

SEDGEWICK, R.; WAYNE, K. **Algorithms**. 4. ed. Boston: Pearson Education, 2011. 955 p.

GOODRICH M. T.; TAMASSIA, R. **Estruturas de Dados & Algoritmos em Java**. Porto Alegre: Bookman, 2013. 700 p.

CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. **Algoritmos – Teoria e Prática**. 3. ed. São Paulo: GEN LTC, 2012. 1292 p.