



Processamento Cosequencial

Cristina D. A. Ciferri

Thiago A. S. Pardo

Leandro C. Cintra

M.C.F. de Oliveira

Moacir Ponti Jr.



Operações Cosequenciais

- Envolvem o **processamento coordenado** (simultâneo) de **duas ou mais listas de entrada sequenciais**, de modo a produzir uma única lista como saída
- Exemplos
 - *merging* (união/intercalação)
 - *matching* (intersecção)
.... de duas ou mais listas ordenadas mantidas em arquivo



Exemplo: como seria o Merging?

| Lista1 | Lista2 |
|---------------|---------------|
| Adams | Adams |
| Carter | Anderson |
| Chin | Andrews |
| Davis | Bech |
| Foster | Rosewald |
| Garwich | Schmidt |
| Rosewald | Thayer |
| Turner | Walker |
| | Willis |

Algoritmo: merging

```
inicializa()      //abre arquivos lista1 e lista2, cria arquivo saída e inicializa  
                  variável existem_mais_nomes como verdade
```

```
lê nome1 da lista1
```

```
lê nome2 da lista2
```

```
enquanto (existem_mais_nomes) faça
```

```
  se (nome1 < nome2)
```

```
    escreva nome1 em saída
```

```
    lê nome1 de lista1
```

```
  senão se (nome1 > nome2)
```

```
    escreva nome2 em saída
```

```
    lê nome2 de lista2
```

```
  senão
```

```
    escreve nome1 em saída
```

```
    lê nome1 de lista1
```

```
    lê nome2 de lista2
```

```
fim-enquanto
```

```
finaliza()       //fecha arquivos
```



Pontos importantes a serem considerados pelo algoritmo

- Inicialização
 - como abrir os arquivos e inicializar as informações para o processo funcionar corretamente
- Sincronização
 - como avançar adequadamente em cada arquivo
- Gerenciamento de condição de fim-de-arquivo
 - processo para ao atingir o fim de uma das listas, sendo que a outra lista continua a ser processada, podendo ser copiada diretamente na saída
- Reconhecimento de erros
 - nomes duplicados ou fora de ordem



Exemplo: como seria o Matching?

| Lista1 | Lista2 |
|---------------|---------------|
| Adams | Adams |
| Carter | Anderson |
| Chin | Andrews |
| Davis | Bech |
| Foster | Rosewald |
| Garwich | Schmidt |
| Rosewald | Thayer |
| Turner | Walker |
| | Willis |

Algoritmo: matching

```
inicializa()      //abre arquivos lista1 e lista2, cria arquivo saída e inicializa
                  variável existem_mais_nomes como verdade
```

```
lê nome1 da lista1
```

```
lê nome2 da lista2
```

```
enquanto (existem_mais_nomes) faça
```

```
  se (nome1 < nome2)
```

```
    lê nome1 de lista1
```

```
  senão se (nome1 > nome2)
```

```
    lê nome2 de lista2
```

```
  senão
```

```
    escreve nome1 em saída
```

```
    lê nome1 de lista1
```

```
    lê nome2 de lista2
```

```
fim-enquanto
```

```
finaliza()       //fecha arquivos
```



Pontos importantes a serem considerados pelo algoritmo

- Inicialização
 - como abrir os arquivos e inicializar as informações para o processo funcionar corretamente
- Sincronização
 - como avançar adequadamente em cada arquivo
- Gerenciamento de condição de fim-de-arquivo
 - processo para ao atingir o fim de uma das listas, sendo que uma lista acabar, encerra-se o processo
- Reconhecimento de erros
 - nomes duplicados ou fora de ordem



Multiway Merging: merging de mais de 2 listas

- Algoritmo

- procura o **menor nome das listas** e escreve
- **lê novo nome** da lista correspondente

- Características

- se existirem **nomes repetidos**, todas as listas devem ser sempre percorridas
- se **não** existirem **nomes repetidos**, pode-se realizar otimizações



Multiway Merging: merging de mais de 2 listas

- Algoritmo (com repetições)

```
...
enquanto (existem_mais_nomes) faça
    menor_nome=min(nome1,nome2,...,nomeK)
    escreva menor_nome em saída
    se (nome1=menor_nome)
        lê nome1 de lista1
    ...
    se (nomeK=menor_nome)
        lê nomeK de listaK
fim-enquanto
...
```



Multiway Merging de Muitas Listas

- Uso de **árvore de seleção**
 - como uma “árvore de torneio”, que guarda a menor das chaves
- Menor chave
 - sempre está na **raiz da árvore**,
 - fácil recuperação
 - deve indicar qual a lista de origem, para se ler novamente da lista correspondente e reestruturar a árvore



Multiway Merging de Muitas Listas

- Exemplo com $K=8$ listas de números

7, 10, 17, ...

9, 19, 23, ...

11, 13, 32, ...

18, 22, 24, ...

12, 14, 21, ...

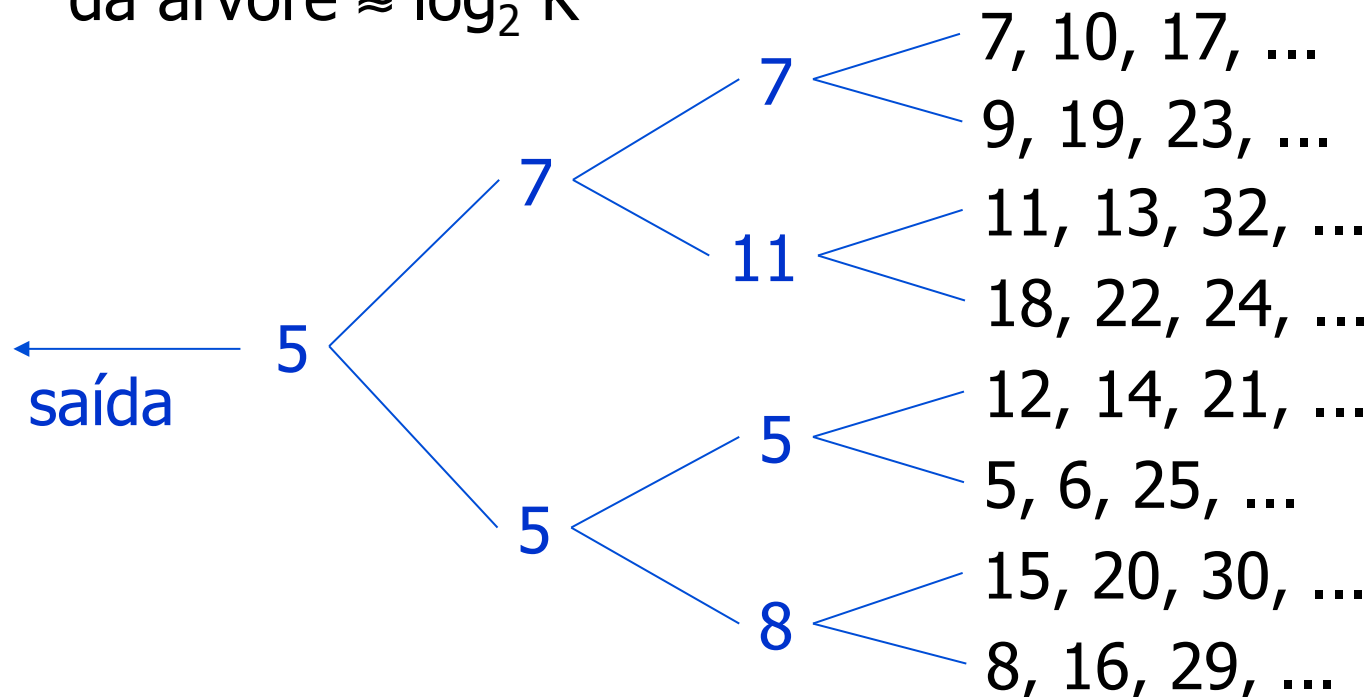
5, 6, 25, ...

15, 20, 30, ...

8, 16, 29, ...

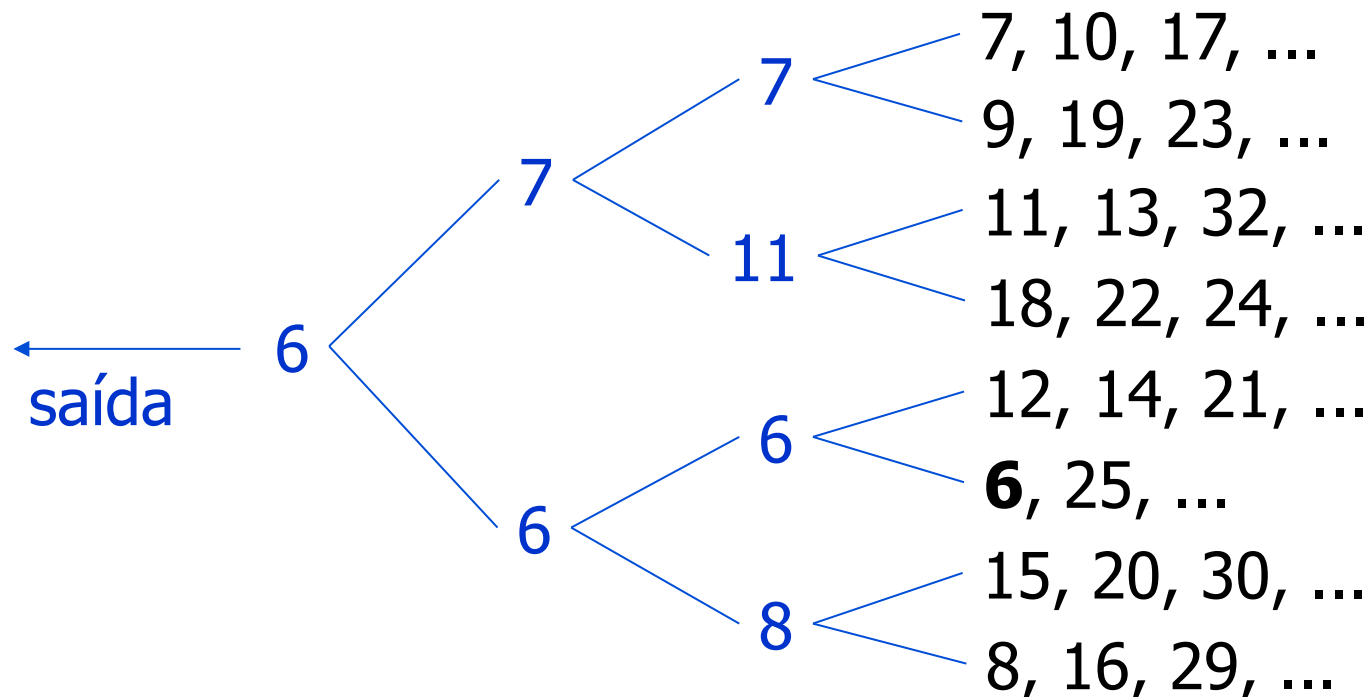
Multiway Merging de Muitas Listas

- Exemplo com $K=8$ listas de números
 - Número de níveis da árvore $\approx \log_2 K$



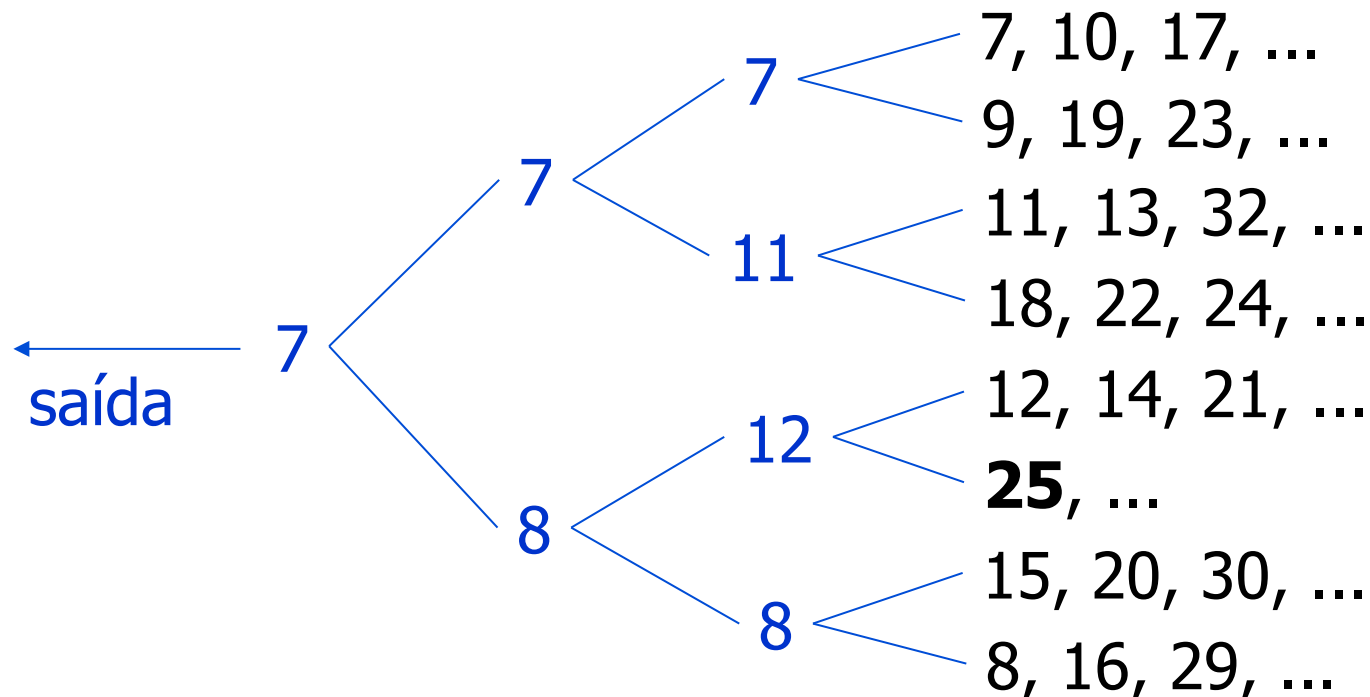
Multiway Merging de Muitas Listas

- Exemplo com $K=8$ listas de números
 - Após processar o 5



Multiway Merging de Muitas Listas

- Exemplo com $K=8$ listas de números
 - Após processar o 6



Multiway Merging de Muitas Listas

- Exemplo com $K=8$ listas de números
 - Após processar o 7

