**Classificação (Ordenação) Externa**

**Classificação**: Esta operação é útil para exibição dos dados de forma mais amigável para usuários de computadores e/ou para tornar outras tarefas mais eficientes.

Quando a **classificação** é aplicada sobre dados residentes em **memória principal**, esta é denominada **classificação interna**. Quando em **memória auxiliar**, **classificação externa**.

As características do meio onde estão armazenados os dados a serem classificados afetam significativamente o processo de ordenação.

Dois principais aspectos dos meios de armazenamento exercem influência sobre a classificação: (1) a **capacidade de armazenamento e (2) velocidade de processamento**.

**Como efetuar classificação Externa?**

Para efetuar a classificação externa, deve-se transferir os dados para memória interna e aplicar métodos de classificação interna ou deve-se efetuar a classificação externa, aplicando em memória externa os mesmos métodos aplicados em memória interna?

A ordenação externa consiste em ordenar arquivos de tamanho maior que a memória interna disponível.

* Os métodos de ordenação externa são muito diferentes dos de ordenação interna.
* Na ordenação externa os algoritmos devem diminuir o número de acesso as unidades de memória externa (custo de acesso).
* Nas memórias externas, os dados são armazenados como um arquivo seqüencial.
* Apenas um registro pode ser acessado em um dado momento.
* Logo, os métodos de ordenação interna são inadequados para ordenação externa.
* Técnicas de ordenação completamente diferentes devem ser utilizadas.

**Fatores que determinam as diferenças das técnicas de ordenação externa**

1. Custo para acessar um item é algumas ordens de grandeza maior.

**2. O custo principal na ordenação externa é relacionado a transferência de dados entre a memória interna e externa.**

3. Existem restrições severas de acesso aos dados.

4. O desenvolvimento de métodos de ordenação externa é muito dependente do estado atual da tecnologia.

5. A variedade de tipos de unidades de memória externa torna os métodos dependentes de vários parâmetros. Assim, apenas métodos gerais serão apresentados.

**Método - Intercalação**

* O método mais importante é o de ordenação por intercalação.
* Intercalar significa combinar dois ou mais blocos ordenados em um único bloco ordenado.
* A intercalação é utilizada como uma operação auxiliar na ordenação.
* **Estratégia geral dos métodos de ordenação externa:**

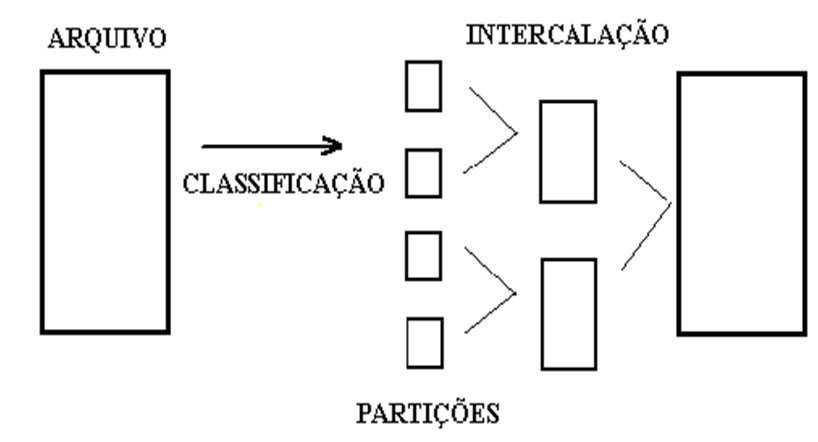
1. Quebre o arquivo em blocos do tamanho da memória interna disponível;

2. Ordene cada bloco na memória interna (RAM);

3. Intercale os blocos ordenados, fazendo várias passadas sobre o arquivo;

4. A cada passada são criados blocos ordenados cada vez maiores, até que todo o arquivo esteja ordenado;

**Modelo – Classificação Externa**



**Revisão: Merge Sort (Ordenação Interna)**

Sua ideia básica consiste em **Dividir** (o problema em vários sub-problemas e resolver esses sub-problemas através da recursividade) e **Conquistar** (após todos os sub-problemas terem sido resolvidos ocorre a conquista que é a união das resoluções dos sub-problemas).

Os três passos úteis dos algoritmos dividir-para-conquistar, ou *divide and conquer*, que se aplicam ao *merge sort* são:

1. **Dividir**: Dividir os dados em subsequências pequenas;
2. **Conquistar**: Classificar as duas metades recursivamente aplicando o *merge sort*;
3. **Combinar**: Juntar as duas metades em um único conjunto já classificado.



**Merge Sort Externo**

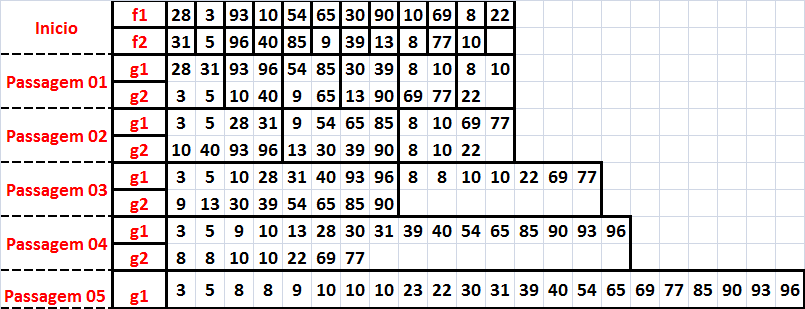
* A versão básica do algoritmo opera com 04 arquivos:
  + Para discutir a idéia do algoritmo, vamos inicialmente assumir que todos os 04 arquivos são armazenados em um único disco;
* Os registros são lidos de **02 arqs. de origem** e reescritos de forma parcialmente ordenada em **02 arqs. de destino**;
* Os arquivos de origem e destino se alternam nas sucessivas iterações do algoritmo;
* Utiliza-se o conceito de **rodada** (run): Subconjuntos ordenados de registros;
* Inicialmente divide-se o arquivo original em dois arquivos *f*1 e *f*2 , ditos **de origem**;
* O número de rodadas de *f*1 e *f*2 (incluindo eventual **cauda**) difere em no máximo 1;
* No máximo um dentre *f*1 e *f*2 possui uma cauda;
* Aquele com cauda possui pelo menos tantas rodadas quanto o outro;
* Uma **cauda** é uma rodada com número incompleto de registros;

**Exemplo**: Rodada de Tamanho 3 e Cauda (5 12).

*f*1: 7 15 29 | 8 11 13 | 16 22 31 | 5 12

*f*2: 8 19 54 | 4 20 33 | 00 10 62 |

* Em princípio vamos tratar os arquivos como estando organizados em rodadas de **tamanho unitário**;
* Inicia-se então a leitura de blocos de registros dos arquivos;
* Por hora considera-se que conjuntos de registros são lidos seqüencialmente de ambos os arquivos e intercalados (**merging por rodadas**);
* Isso significa que cada rodada de um arquivo é fundida com a rodada correspondente do outro arquivo, **formando uma rodada com o dobro do tamanho**. **Exemplo:**



**Conclusão: O método envolve 02 fases: Geração das rodadas (blocos ordenados de dados) e intercalação.**

**Intercalação – Balanceada em Vários Caminhos (T-Vias)**

**Passo 01**: Particionar em blocos de forma equilibrada (baseado no tamanho da memória);

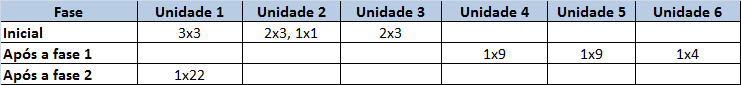
**Passo 02**: Ordena os blocos;

**Passo 03**: Distribuição equitativa dos blocos nas unidades;

**Passo 04**: Intercalar os blocos compondo blocos maiores (em unidades de armazenamento VAZIA);

A capacidade de registros (tamanho de blocos) da memória interna e o número de unidades devem ser informados.

**Exemplo**: Material (Kenia)



**Exercício 01:** Quantas fases são necessárias para ordenar **2.000 registros** considerando blocos de tamanho 100 em 04 unidades (sendo 02 de entrada e 02 de saída). Mostre a distribuição dos blocos nas unidades em todas as fases.

**Intercalação – Polifásica**

Os blocos são distribuídos entre unidades de forma desigual e somente **UMA** unidade é reservada para armazenar os blocos resultantes das intercalações. A cada passo, uma unidade deve esvaziar-se. A intercalação é concuída até que uma das unidades esvazie-se.

* A intercalação é realizada em muitas fases;
* As fases não envolvem todos os blocos;
* É ligeiramente melhor do que a intercalação balanceada;

**Exemplo**: Material (Kenia)

