

4ª Lista de Exercícios

1. O desempenho de um processo de ordenação interna é, geralmente, medido em função do número de comparações necessárias. Explique por que o número de comparações não é uma medida adequada para avaliar o desempenho de métodos de ordenação de grandes arquivos que não cabem em memória.
2. Escreva um algoritmo que faça o seguinte:
 - a) Examine o conteúdo de dois arquivos ordenados, M1 e M2;
 - b) Produza um terceiro arquivo chamado COMUM contendo a cópia dos registros que existam nos dois arquivos (AND);
 - c) Produza um quarto arquivo chamado DIFERENTE contendo os registros dos dois arquivos que existam em apenas M1 ou M2 (XOR).
3. Suponha que um arquivo de dados com 6.000 registros, mantido em disco, deve ser ordenado em um computador cuja memória interna acomoda no máximo 600 registros por vez, usando o procedimento de *Merge Sort* visto em aula. Essa área de memória interna é usada como buffer de entrada para leitura de dados do disco e o sistema conta com um *buffer* de saída adicional que acomoda 200 registros.
 - a) Durante o *merging*, quantos registros serão lidos de cada corrida cada vez que ela é acessada? Justifique.
 - b) Quantos *seeks* serão realizados para a leitura dos registros durante a etapa de *merging* (excluindo a fase de geração de corridas)?
 - c) Quantos *seeks* serão realizados para escrever o arquivo ordenado no disco? Justifique.
4. O *Merge Sort* em múltiplos passos requer mais processamento interno e operações de E/S extras. Então por que, em geral, é mais rápido do que o *Merge Sort* em passo único?
5. Considerando o arquivo de 800 MB usado como exemplo nas aulas, um *buffer* de entrada de 4 MB e *buffers* de saída de 200 KB, calcule o custo total e de cada fase do *Merge Sort* (em termos de bytes transmitidos e número de *seeks*) quando realizado em dois passos: primeiro 20 x 10-vias, depois 20-vias.
6. Considerando um arquivo de 10 GB, um *buffer* de entrada de 20MB e *buffers* de saída de 250 KB, calcule o custo total e de cada fase do *Merge Sort* (em termos de bytes transmitidos e número de *seeks*) em um passo. Em seguida, calcule o custo do *Merge* quando realizado em dois passos: primeiro 25 x 20-vias, depois 25-vias.