

## MC326 A — Estruturas de Arquivos

**Professor**

Ricardo da Silva Torres

&lt;rtorres@ic.unicamp.br&gt;

**Segundo Semestre de 2006**

Para as questões 1 a 5, a  $j$ -ésima operação é realizada sobre a árvore resultante da  $(j-1)$ -ésima operação. Considere a árvore vazia para  $j = 0$ .

1. Qual a árvore B+ final (ordem 4 para o *index set* e ordem 3 para o *sequence set*), supondo a inserção de chaves na ordem a seguir: 1, 7, 15, 16, 18, 20, 40, 6, 12, 14, 5, 9, 8, 10, 11, 19, 41.
2. A partir da árvore final do exercício anterior, faça as seguintes operações em ordem. Eliminar 18, 15, 6, 20, 8, 9, 10, 12.
3. Repita o exercício 2, considerando agora uma árvore B de ordem 5.
4. Repita o exercício 3 considerando a árvore B final obtida no exercício anterior.
5. Quais as vantagens do uso da técnica de redistribuição quando da inserção e eliminação de chaves em uma árvore B?
6. Considere a árvore *simple prefix B+* (de ordem 2 para o *index set*) mostrada na Figura 1. Suponha que uma chave é adicionada ao bloco 5 resultando em um *split* do bloco 5 e a consequente adição do bloco 8 como mostrado na Figura 2:
  - Desenhe a árvore resultante após esta inserção.
  - Suponha que, após a inserção, uma deleção causa *underflow* e a consequente concatenação dos blocos 4 e 5. Desenhe a árvore resultante após esta deleção.
  - Descreva um caso no qual uma deleção resulta em redistribuição, ao invés de concatenação, e mostre o efeito dessa operação na árvore.
7. Suponha um arquivo de dados com registros de tamanho variável, indexado por um arquivo baseado em chave primária apenas. Considere as seguintes operações: inserção de registro, eliminação de registro e modificação de registro.

Quais destas operações são necessárias, no arquivo de índice, para

  - inserção de um registro no arquivo de dados?
  - eliminação de um registro no arquivo de dados?

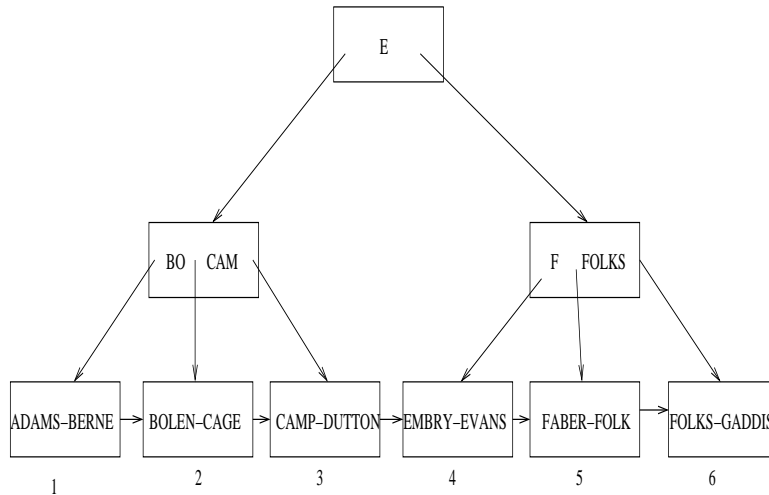


Figura 1:

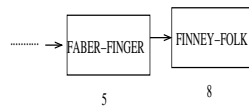


Figura 2:

- modificação de um valor de um campo não chave no arquivo de dados?
8. Seja um arquivo de dados, não ordenado, com registros de tamanho fixo, fator de bloco igual a 10, ocupando 1000 blocos em disco. Os registros são compostos pelos campos RA, NOME, TELEFONE e DNASCIMENTO. Este arquivo é indexado por três índices:
- índice 1 relativo ao campo RA,
  - índice 2 relativo ao campo TELEFONE e
  - índice 3 relativo ao campo NOME;

considere ainda que RA e TELEFONE são chaves primárias (não pode haver repetição) e NOME é chave secundária. Cada índice é independente dos demais (ou seja, cada índice aponta diretamente para o arquivo de dados).

- Quantos registros tem o arquivo de dados? Quantos registros tem o arquivo de índice 1? Quantos registros tem o arquivo de índice 2? Quantos registros tem o arquivo de índice 3? Justifique cada resposta.
- Suponha que os três índices caibam na memória, mas que o arquivo de dados não caiba. Suponha além disso que os quatro arquivos já estejam abertos e que os índices já estejam na memória, que tem lugar para mais alguns blocos de dados.

Considere as seguintes operações, onde Arq é o nome de um dos arquivos (Dados, Índice 1, Índice 2, Índice 3): Read(Bloco, Arq), Write(Bloco, Arq), Insert(Registro, Arq), Delete(Registro, Arq), Update(Registro, Arq).

Exemplo: Delete(Registro, Dados) indica que um registro foi eliminado do arquivo de dados.

Quais destas operações precisam ser efetuadas para

- a) Modificar DNASCIMENTO de uma pessoa no arquivo de dados?
- b) Inserir um registro no fim do arquivo de dados? Neste caso de inserção, para cada operação mencionada, diga o número mínimo de operações de I/O que devem ser realizadas, explicando em cada caso o porquê deste número.

9. Considere o mesmo arquivo de dados da pergunta anterior. Os três índices agora não são mais independentes, mas organizados da seguinte forma:

- índice 1 (RA) aponta diretamente para o arquivo de dados,
- índice 2 (TELEFONE) aponta para o índice 1 e
- índice 3 (NOME) aponta para o índice 2.

Suponha que os três índices caibam na memória, mas que o arquivo de dados não caiba. Suponha, além disso que os quatro arquivos já estejam abertos e que os índices já estejam na memória, que tem lugar para mais alguns blocos de dados.

Considere as mesmas operações, onde Arq é o nome de um dos arquivos (Dados, Índice 1, Índice 2, Índice 3): Read(Bloco, Arq), Write(Bloco, Arq), Insert(Registro, Arq), Delete(Registro, Arq), Update(Registro, Arq).

Exemplo: Delete(Registro, Dados) indica que um registro foi eliminado do arquivo de dados.

- Quais destas operações precisam ser efetuadas para inserir um registro no fim do arquivo de dados?
- Suponha agora que nenhum índice cabe na memória, e tampouco o arquivo de dados. Suponha que os quatro arquivos já estejam abertos e que os índices estejam parcialmente na memória, bem como parte do arquivo de dados. Considere as seguintes afirmações:
  - a) Agora, para inserir um registro no fim do arquivo de dados serão necessárias sempre mais operações de I/O do que no caso do item anterior. CERTO ou ERRADO? Justifique.
  - b) Agora, para modificar o NOME de uma pessoa, como o arquivo de dados tem registros de tamanho fixo, basta fazer I/O nos arquivos de dados e índice 3. CERTO ou ERRADO? Justifique.

10. Considere um índice (árvore B) com 1MB de registros e a disponibilidade de apenas 256KB de RAM para armazená-lo. Dado que o tamanho de uma página do índice é 4KB, com 64 chaves por página; quantos acessos são necessários para recuperar um dado no pior caso? O que fazer para melhorar o número médio de acessos ao disco?

11. Dada uma árvore B de ordem  $m$  com  $N$  chaves, encontre uma fórmula para uma busca no pior caso? E no melhor caso?

12. Escreva um algoritmo em alto nível para a deleção em árvores B+.
13. Quais os problemas de se usar árvores binárias de busca e árvores AVL como estruturas de índices?
14. Seu “patrão” quer construir uma árvore B com páginas de 512 bytes para indexar um arquivo da empresa. Ele tem 1000 chaves associadas a informações genéricas. Suponha que o tamanho necessário para armazenar uma chave e suas informações associadas é 64 bytes. Além disso, ele sabe que se armazenar as informações em um outro arquivo, ele pode apenas manter a chave e um ponteiro para o arquivo de informações em apenas 16 bytes. O que ele deve escolher? Armazenar as informações junto a chave ou a guardar apenas um ponteiro para um outro arquivo? Na sua análise, considere apenas a realização de buscas.
15. Dada uma árvore B de ordem 512:
  - Qual é o máximo número de descendentes de uma página?
  - Qual é o mínimo número de descendentes de uma página (excluindo a raiz e folhas)?
  - Qual é o mínimo número de descendentes do nó raiz?
  - Qual é o mínimo número de descendentes de uma folha?
  - Quantas chaves há numa página não-folha com 200 descendentes?
  - Qual é a máxima profundidade da árvore se ela contém 100000 chaves?
16. Descreva como prover acesso sequencial ordenado a um arquivo indexado por uma árvore B.
17. Quais as vantagens da árvore B+ em relação à árvore B?
18. Considere um arquivo onde os seguintes valores são chaves para uma função de hash: 2369, 3760, 4692, 4871, 5659, 1821, 1074, 7115, 1620, 2428, 3943, 4750, 6975, 4981, 9208. O arquivo de hash usa oito *buckets*, numerados de 0 a 7. Cada *bucket* é um bloco de dois registros. Carregue estes registros no arquivo na ordem dada, usando a função de *hash*  $h(K) = K \bmod 8$ . Calcule o número médio de acessos aos blocos para uma recuperação aleatória.
19. A tabela seguinte mostra a distribuição das chaves resultantes de diferentes funções de hash em um arquivo com 6000 registros e 6000 endereços.

	Função A	Função B	Função C
d(0)	0.71	0.25	0.40
d(1)	0.05	0.50	0.36
d(2)	0.05	0.25	0.15
d(3)	0.05	0.00	0.05
d(4)	0.05	0.00	0.02
d(5)	0.04	0.00	0.01
d(6)	0.05	0.00	0.01
d(7)	0.00	0.00	0.00

- Quais funções (se existir) geram um distribuição de registros que é aproximadamente aleatória?

- Qual função gera uma distribuição próxima a uniforme?
- Quais funções (se existir) geram uma distribuição que é pior que a aleatória?
- Qual função deveria ser escolhida?

## Referências

- [1] M. J. Folk and B. Zoellick. *File Structures*. Addison-Wesley, 1992.