

Atividade: Laboratório 04– B Tree

1. Objetivo: Implementar em C um conjunto de rotinas para:

- a construção de um índice de acesso a um arquivo de dados utilizando Árvores B
- a pesquisa de informações utilizando a Btree construída

2. Problema: Modificar parte do laboratório 2

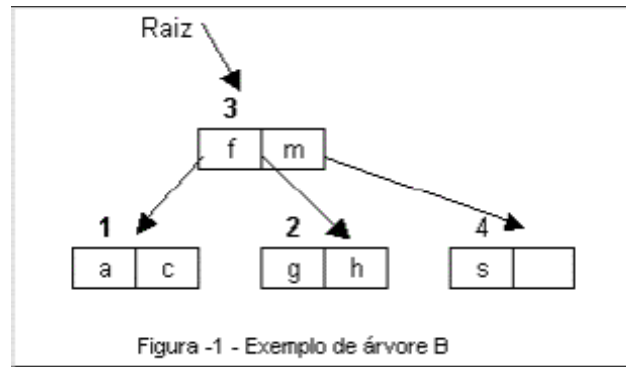
- Em vez de gerar o índice em disco, para pesquisa binária em disco, criar uma Árvore B em disco.
- Use o arquivo de entrada do laboratório 2 (Processamento sequencial) como arquivo de dados.
- É **imprescindível** que seja estudado o capítulo de Árvores B do livro texto.
- O campo chave é a matrícula do aluno
- Reutilizem tudo que for possível dos laboratórios anteriores: configurações, rotinas, etc..

3. Saída 1 - Arquivo de índices da chave primária usando B-TREE.

- O arquivo de índices deverá conter a chave e um apontador que permita localizar o registro no arquivo de dados.
- **IMPORTANTE:** os registros de dados NÃO devem ser armazenados nos nós da árvore.
- Caso o arquivo de dados apresente chaves duplicadas, considerar a primeira ocorrência como chave correta e as próximas ocorrências devem ser descartadas. Os registros descartados devem ser gravados em um arquivo de “descartados”. O formato dos registros desse arquivo é o mesmo que do arquivo de entrada.
- A ordem da B-Tree é um parâmetro a ser fornecido pelo usuário, variando de 3 a 10 (inclui os extremos).
- Caso facilite sua vida, você pode definir estruturas de dados de tamanho fixo, que possibilitem árvores de ordem de 3 a 10. (Por exemplo, no caso de uma árvore de ordem 3, não tem problema que fique sobrando memória nos nós da árvore).

4. Saída 2 - Arquivo texto que descreva a estrutura da B-tree criada

- Seu programa deverá também gerar um arquivo texto com a estrutura da B-Tree criada contendo informações suficientes que permitam ao usuário “desenhar” no papel toda a árvore. Para a B-tree da Figura-1, de ordem 3, vamos usar a seguinte convenção:
 1. Os apontadores de filhos, de uma chave que não tenha filhos, serão representados com endereço -1.
 2. O par (n,X) deve ser interpretado como: “Uma chave menor que **X** está na sub-árvore de endereço **n**”;
 3. O par (k,) deve ser interpretado como: “Uma chave **maior do que todas as chaves do nó** está na sub-árvore **k**”
 4. Os endereços dos nós da B-tree estão indicados com números de 1,2,3,4 para facilidade de entendimento. Eles podem ser os endereços do disco onde está gravado nó.
- .



Voce pode escolher a saída a ser gerada entre as três opções a seguir.

- Opção 1

Raiz: 3
 Nível 1 da árvore:
 Nó 3 : Total chaves=2; (1,f),(2,m),(4,)
 Nível 2 da árvore:
 Nó 1 : Total chaves=2; (-1,a), (-1,c), (-1,)
 Nó 2 : Total chaves=2; (-1,g), (-1,h), (-1,)
 Nó 3 : Total chaves=1; (-1,s), (-1,)

- Opção 2

Raiz: 3
 Nó 3 : Total chaves=2; (1,f),(2,m),(4,)
 Nó 1 : Total chaves=2; (-1,a), (-1,c), (-1,)
 Nó 2 : Total chaves=2; (-1,g), (-1,h), (-1,)
 Nó 3 : Total chaves=1; (-1,s), (-1,)

- Opção 3

1. Voce pode propor uma outra forma de gerar as informações que permitam reconstruir a árvore.
2. Use seus conhecimentos para criar diretamente o “desenho” da árvore;

6. Ativação do programa

Nome do programa: Btree

Chamada: Seu programa deverá ser chamada via linha de comando com a seguinte sintaxe

Btree <ordem Btree> <arqdados> <arqindice> <Arqdespreza> <Arqdesc>

- **Parâmetro Descrição**

- <Ordem Btree> - Ordem da Btree. Valor entre 3 e 10, inclusive extremos.
- <Arqdados> - Nome do arquivo de dados
- <Arqindice> - Nome do arquivo de índices
- <Arqdespreza> - Nome do arquivo com as chaves desprezadas
- <Arqdesc> - Nome do arquivo com a descrição da estrutura da árvore que permitirá reconstruí-la graficamente.

7. Menu de Opções do programa

Ao ser ativado o programa deverá ficar em “loop” mostrando as opções abaixo, até que seja pedido o seu encerramento.

1. Construir a Btree
2. Imprimir/Gerar um arquivo com a estrutura da Btree
3. Pesquisar a base de dados (OPCIONAL)
4. Encerrar o programa
- 5.

8. Preparação dos testes

1. Crie arquivos de teste com 1 registro, 3 registros, 20 registros, 10.000 registros
2. Para cada arquivo acima, teste seu programa para árvores de ordem 2,3,5,7

9. Livro Texto

1. Folk, Michael J.; Zoellick, Bill; Riccardi Greg - File Structures: an object oriented approach with C++ ; editora Addison Wesley, 1998
2. Folk, Michael J.; Zoellick, Bill; Riccardi Greg - File Structures 2nd ed Addison Wesley, 1992, 590 pp
3. <http://www.icmc.sc.usp.br/manuals/sce183/btree2.html>
4. <http://www.semaphorecorp.com/btp/algo.html>

5.

Data Entrega	Valor	
20/06 – até 23 horas	10,00 (sem opcional)	
23/06 – até 21 horas	10,00 (com opcional) 9,00 (com opcional)	
25/06 e 29/06		Demonstrar o programa no laboratório