

2º Trabalho Prático

O programa referente ao 2º trabalho prático da disciplina Organização e Recuperação de Dados deverá ser desenvolvido na linguagem C por grupos de até três pessoas. O código-fonte deverá estar devidamente identificado com os nomes das pessoas do grupo e ser entregue exclusivamente via *Classroom*. Posteriormente, o grupo deverá fazer a apresentação do trabalho em horário previamente agendado.

Especificação

O objetivo deste trabalho é criar um programa que, a partir de um arquivo contendo uma lista de chaves, construa uma **árvore-B**. Além disso, o programa também deverá imprimir informações sobre a árvore construída que permitam avaliar como a árvore-B se comporta dadas sequências de chaves de tamanhos e ordenações diferentes.

A geração do arquivo de chaves será realizada por um programa auxiliar que será fornecido juntamente com esta especificação. O arquivo de chaves será um arquivo texto contendo uma chave por linha e as chaves serão números inteiros. **Dica:** você pode ler as chaves do arquivo de chaves como números inteiros utilizando a função `fscanf()`.

Cada chave lida deverá ser inserida em uma árvore-B de **ordem m** , sendo que o valor de m será definido em tempo de compilação. Para facilitar a experimentação com árvores de diferentes ordens, faça m ser uma constante e o use-a ao longo do código sempre que precisar referenciar a ordem da árvore.

A estrutura interna das páginas da árvore de ordem m deve ser similar à vista em aula. **A árvore-B deverá ser criada e mantida em arquivo**, logo, a árvore nunca deverá estar **completamente** carregada na memória.

O programa deverá oferecer as seguintes funcionalidades:

- Criação da árvore-B a partir do arquivo de chaves;
- Impressão das informações da árvore-B.

Assim como no 1º trabalho, as operações serão realizadas sem interação direta com o usuário, utilizando para isso a linha de comando com passagem de parâmetros.

Criação da árvore-B

A funcionalidade de criação da árvore-B (`-c`) deve ser acessada pela linha de comando, da seguinte forma:

```
$ programa -c nome_arquivo_chaves
```

sendo `programa` o nome do arquivo executável do seu programa e `nome_arquivo_chaves` o nome do arquivo que contém as chaves. Sempre que ativada, essa funcionalidade fará a leitura do arquivo de chaves e a inserção das chaves na árvore-B, que será armazenada no arquivo `btree.dat`. Se o arquivo `btree.dat` já existir, ele deverá ser reescrito. Note que o formato do arquivo de chaves será sempre igual, porém o número de chaves no arquivo pode variar. Para simplificar o processamento do arquivo de chaves, considere que ele sempre será fornecido corretamente (i.e., o seu programa não precisa verificar a integridade desse arquivo).

Ao final da criação da árvore-B, o programa deverá apresentar uma mensagem na tela indicando se essa operação foi concluída com sucesso ou não.

Impressão das informações da árvore-B

A funcionalidade de impressão das informações da árvore-B (`-p`) também será acessada via linha de comando, no seguinte formato:

```
$ programa -p
```

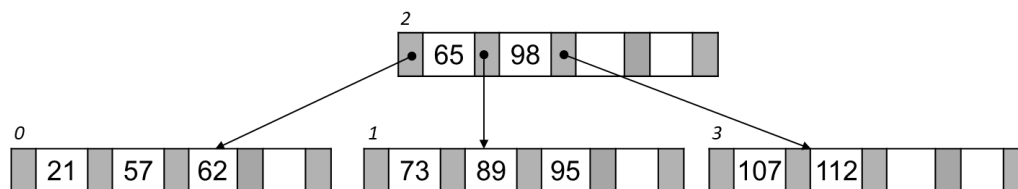
sendo `programa` o nome do arquivo executável do seu programa. Note que, para essa execução, o arquivo `btree.dat` deve existir. Caso o arquivo não exista, o programa deve apresentar uma mensagem de erro e terminar.

Sempre que ativada, essa funcionalidade apresentará na tela o conteúdo de todas as páginas da árvore-B armazenada no arquivo *btree.dat*. Para cada página da árvore deverá ser informado: (a) o seu RRN; (b) os valores do vetor de chaves; e (c) os valores do vetor de filhos. As páginas devem ser apresentadas pela ordem do seu RRN e a página raiz deve ser devidamente identificada. Ao final, as seguintes informações devem ser apresentadas:

- Altura da árvore-B;
- Número de chaves da árvore-B;
- Número de páginas da árvore-B;
- Taxa de ocupação da árvore-B.

A *altura da árvore-B* é um número inteiro que indica quantos níveis a árvore-B possui. Note que essa informação permite estimar o número máximo de leituras necessárias para se localizar uma chave qualquer. O *número de chaves e de páginas da árvore-B* também são números inteiros e representam o número total de chaves inseridas e de páginas criadas, respectivamente. Por fim, a *taxa de ocupação da árvore-B* representa a porcentagem de ocupação média de suas páginas. Este valor pode ser calculado dividindo-se a quantidade de chaves armazenadas na árvore pela quantidade de espaço reservado a elas pelas páginas.

Como um exemplo, considere a árvore-B construída a partir da inserção das seguintes chaves: 65 21 57 89 73 62 98 112 107 95 (nesta ordem).



Supondo que esta seja a árvore construída, seu programa deverá apresentar as seguintes informações:

```

RRN: 0
Chaves: 21 | 57 | 62
Filhos: -1 | -1 | -1 | -1

RRN: 1
Chaves: 73 | 89 | 95
Filhos: -1 | -1 | -1 | -1

- - - - Pagina Raiz - - - -
RRN: 2
Chaves: 65 | 98
Filhos: 0 | 1 | 3
- - - - -

RRN 3:
Chaves: 107 | 112
Filhos: -1 | -1 | -1
- - - - -

Estatísticas da Arvore-B
> Altura: 1
> Numero de chaves: 10
> Numero de paginas: 4
> Taxa de ocupacao: 62.50%
  
```