

## Trabalho 02

**Disciplina:** *Organização de Arquivos – Turma C*

**Semestre:** 2º/2016

**Prof.:** André Drummond

**Título:** *Árvores-B*

**Entrega:** *05/12/2016 – 23h50 (Arquivos-Fontes via Aprender.unb.br, num único arquivo zipado. Incluir um comentário com os nomes e matrículas dos alunos. O arquivo zipado deve ser a matrícula do líder do grupo, ex.: 0912345.zip )*

### 1. Objetivos

O objetivo deste trabalho é que o aluno aprenda a trabalhar com a estrutura de dados árvore-B para indexação de arquivos.

Implementar, utilizando obrigatoriamente a linguagem C ou C++.

### 2. Especificação

O arquivo *data1.txt* disponibilizado no Moodle contém registros de tamanho variável. Onde cada campo é separado por ";". Os campos são: *Chave Primária, Nome, Sobrenome, Empresa, Endereço, Cidade, Estado, ZIP/Postal Code, Phone 1 e Phone 2.*

O arquivo *data2.txt* contém registros de tamanho fixo onde o primeiro campo é a chave primária, os outros campos são dados.

A partir desses arquivos crie o programa 'btree.c'. O programa deve receber como argumento por linha de comando o nome de um arquivo e a opção '-r 1' ou '-r 2' indicando se o arquivo é de registros de tamanho variável ou fixo, respectivamente. Exemplo:

```
btree data1.txt - r 1
```

O programa deve abrir os arquivos de dados e gerar uma árvore B de **ordem 5** com os índices primários (primeiro campo do registro) e a respectiva PRR ou NRR dos registros. Depois, o programa deve fechar os arquivos de dados e mostrar o seguinte menu:

- 1 - Buscar Registro
- 2 - Inserir Registro
- 3 - Mostrar árvore-B

## A árvore-B deve sempre ficar em memória principal NÃO É NECESSÁRIO ESCREVER A ÁRVORE-B NUM ARQUIVO.

Se a **opção 1** for escolhida deve pedir para o usuário a chave primaria a ser buscada, buscar na árvore-B, e mediante a NRR ou PRR abrir o arquivo de dados e ler diretamente o registro desejado e mostrar na tela. Mostrar também quantos SEEKS seriam necessários para achar o registro, assumindo que a árvore-B se encontra em memória secundaria.

Se a **opção 2** for escolhida deve-se pedir ao usuário todas as informações do registro e colocar no final do arquivo de dados. Além disso, deve inserir de forma correta o índice na árvore-B.

Se a **opção 3** for escolhida deve-se mostrar a árvore-B no seguinte formato:

Nx: Px: CP1|CP2|CP3|CP4 p1|p2|p3|p4|p5 ; Px': CP1|CP2|CP3|CP4 p1|p2|p3|p4|p5

Onde Nx - indica o nível da árvore, Px: indica o número da pagina, CPx - indica chave primaria e px indica ponteiro para página filho. No caso, CPx deve ser substituído pelo valor da chave primaria do registro na posição x da página. E px deve ser substituído pelo número da página filha, quando for uma folha deve ser mostrado o caractere '-'. Exemplo:

N0: P3: ID23 0|1  
N1: P0: ID12|ID15|ID24 -|-; P1: ID25|ID28| -|-

## 3. Grupos

Neste projeto será permitido a formação de grupos de no máximo 2 alunos. A partir do grupo formado, deverá ser indicado um líder que será o responsável pelo envio dos arquivos fontes para o sistema *Aprender.unb.br*. Somente serão aceitos os arquivos fontes enviado pelo líder do grupo.

## 4. Observações

- Trabalho identificado como cópia entre alunos ou de algoritmos da Internet (qualquer uma das partes) recebe a nota ZERO no trabalho.
- Os trabalhos devem seguir boas práticas de programação: Identação, não declarar variáveis no meio do código, comentários, etc.
- Não serão aceitos trabalhos atrasados BAIXO NENHUMA HIPÓTESE.