



UNIVERSIDADE DE BRASÍLIA  
-UnB  
Departamento de Ciência da Computação (CIC)  
Curso de Licenciatura em Computação

JOÃO PAULO VAZ MENDES  
17/0002934

# PROJETO II DE ESTRUTURA DE DADOS

BRASÍLIA  
2017

## RELATÓRIO

Primeiramente para se executar o programa deve se extrair o conteúdo do arquivo '170002934\_JoaoMendes.zip', abrir o terminal de comando do sistema operacional e com o comando 'cd' se dirigir ao diretório onde se extraiu o arquivo: (ex: \$ cd Documentos).

Para garantir que o arquivo com os dados a serem lidos seja aberto e lido corretamente deve se certificar que o arquivo '.csv' esteja contido na mesma pasta que o arquivo de execução.

A seguir deve-se compilar o arquivo principal e o arquivo de implementação das funções que foi realizado de acordo com as diretrizes de Tipo Abstrato de Dados (TAD). A compilação então deve ser feita a partir do compilador gcc. Recomendo que ela seja realizada com o comando a seguir:

**\$ gcc main.c bib.c -o main**

Deste modo o programa será compilado e o executável gerado será o 'main' o qual eu usarei neste relatório.

A seguir execute o programa no mesmo diretório pelo comando no terminal:

**\$ ./main**

Depois de executado, surgirá dados confirmando a montagem de cada uma das três árvores de busca balanceadas. Sendo cada uma identificada pelo método usado de chave e apresentando a quantidade de nós totais, quantidade de nós da sub-árvore esquerda, quantidade de nós da sub-árvore direita e a altura da árvore.

```
Árvore com nota de postura profissional media como chave criada!  
número de nós da abb1: 1499  
número de nós esquerdo da abb1: 663  
número de nós direito da abb1: 835  
altura da abb1: 13
```

A seguir surgirá no terminal o menu de consultas do programa:

```
Ambiente de consultas das avaliações de docentes - Opcoes  
(1) Mostrar informações do nó raiz  
(2) Mostrar a média das chaves das sub-árvores esquerda e direita  
(3) Mostrar todas as avaliações de um docente para as 3 chaves  
(4) Pesquisar o nome dos docentes da árvore esquerda com um das 3 chaves  
(5) Pesquisar o nome dos docentes da árvore direita com um das 3 chaves  
(6) Apresentar a quantidade de vezes que um docente aparece na árvore direita com uma das 3 chaves  
(7) Apresentar a quantidade de vezes que um docente aparece na árvore direita com uma das 3 chaves  
(0) SAIR  
Digite a sua opcao:
```

Este menu pede para que se escolha uma opção de consulta.

Por exemplo, a consulta número 1 do menu. Basta inserir o 1 como operação desejada:

```
Digite a sua opcao: 1
```

```
Informações do nó raiz da árvore 1 - chave: postura profissional media
```

```
chave: 9.430000  
código do docente: 15370  
nome do docente: FERNANDO MOREIRA DA SILVA  
turma: 1241962  
ano: 2015  
periodo: 1  
quantidade de discentes: 19  
postura profissional média: 9.430000  
postura profissional DP: 1.160000  
atuação profissional média: 6.680000  
atuação profissional DP: 3.480000  
autoavaliação aluno média: 8.550000  
autoavaliação aluno DP: 1.680000
```

```
Informações do nó raiz da árvore 2 - chave: atuação profissional media
```

```
chave: 9.270000  
código do docente: 15370  
nome do docente: FERNANDO MOREIRA DA SILVA  
turma: 57574771  
ano: 2016  
periodo: 1  
quantidade de discentes: 8  
postura profissional média: 9.750000  
postura profissional DP: 0.450000  
atuação profissional média: 9.270000  
atuação profissional DP: 1.110000  
autoavaliação aluno média: 9.070000  
autoavaliação aluno DP: 1.540000
```

```
Informações do nó raiz da árvore 3 - chave: autoavaliação aluno media
```

Outro exemplo pode ser dado com a opção 5, onde depois de digitada a opção deve se digitar o código do docente que deseja procurar, depois o número da árvore onde deseja pesquisar. Então o programa irá retornar o resultado.

```
Digite a sua opcao: 5
```

```
Digite o código do professor: 15373
```

```
Escolha uma árvore para busca (1, 2 ou 3): 2
```

```
O professor está presente na sub-árvore direita
```

Depois de encerrada a consulta basta escolher a opção 0 para sair do programa.

## Construção da Árvore Binária de Busca Balanceada

Primeiramente é definida uma estrutura do nó da árvore preenchida com os dados contidos em cada, sendo essencial uma chave de busca, um ponteiro para o nó da esquerda e um ponteiro para o nó da direita, ambos sendo variáveis do mesmo tipo da estrutura que pertencem com um ponteiro.

Depois no arquivo 'bib.h' defino o tipo de dados da estrutura nó como sendo 'Tree'.

A árvore é inicializada a partir da função "Tree \*\* cria\_inicializa\_Tree()" que retorna o tipo ponteiro para facilitar o trabalho com a raiz da árvore. Nesta função se realiza a alocação dinâmica do nó raiz e o começo da árvore aponta para 'NULL' por ainda não existir um nó preenchido.

Os nós da árvore são preenchidos pela função "int insere\_Tree(Tree \*\*raiz, Data elem, DadoArvore \*info)". Esta função recebe como argumento a raiz de uma árvore já inicializada, vazia ou não, uma estrutura de elementos de informação do nó e uma estrutura para fornecer informações gerais da árvore.

Nesta função primeiro se confirma se ela está vazia. Se estiver se aloca um novo nó dinamicamente, passam-se as informações para ele e este é passado para a raiz da árvore. No caso de não estar vazia o processo de inserção continua.

Um nó auxiliar é criado a partir da estrutura Tree e este recebe o ponteiro da raiz. A partir de então se comparam a chave da raiz e do novo nó. Caso a chave seja menor ou igual à chave da raiz está prossegue para a sub-árvore esquerda onde iniciará um processo de recursividade até chegar ao final da árvore e ser alocado dinamicamente. Caso contrário o processo prossegue pela sub-árvore direita. Tal processo é feito do mesmo modo para cada nó, onde a chave menor ou igual prossegue para a esquerda e a maior para a direita. Uma vez que o nó é alocado com sucesso, retornando 1 na função, a função "insere\_Tree" chama a função "fatorBalanceamentoNO" a qual constata se o novo nó desbalanceou a árvore fazendo com que uma das sub-árvores tenha a altura maior em no mínimo 2 de altura. Caso isso ocorra esta função 2 ou mais e a árvore será submetida a um processo de rotação.

A rotação depende da sub-árvore que o último nó inserido se encontra e se este está a esquerda ou direita do seu nó pai direto. Caso o nó esteja na sub-árvore esquerda e estiver à esquerda do último nó inserido se realiza uma rotação simples para a esquerda (Rotação LL), no caso deste mesmo estar à direita do último nó inserido se realiza uma rotação dupla esquerda direita (Rotação LR). O caso é semelhante na sub-árvore direita, porém com os lados opostos.

A rotação simples se dá de modo que é criado um nó auxiliar direcionado para ser um dos lados filhos da raiz. Este mesmo lado filho da raiz serpa depois ocupado com o nó filho oposto do nó auxiliar. E este nó oposto auxiliar será preenchido com a raiz. Após este processo se calcula a altura de cada um dos nós e a raiz toma o lugar do nó auxiliar.

A rotação dupla se dá de modo a realizar duas rotações simples de lados opostos, variando qual é a primeira ou segunda.

Após as devidas rotações onde a altura é calculada e se constata o balanceamento da árvore esta tem sua construção finalizada e pronta para servir como estrutura de busca.