



UFAM

FT – FACULDADE DE TECNOLOGIA
ENGENHARIA DA COMPUTAÇÃO

JEAN CLEISON BRAGA GUIMARÃES – 21601227

TRABALHO 2: ÁRVORE-B
IMPLEMENTAÇÃO DE UM PROGRAMA DE CONTROLE DE FLUXO
FINANCEIRO

MANAUS, AM
2019

JEAN CLEISON BRAGA GUIMARÃES

**TRABALHO 2: ÁRVORE B
IMPLEMENTAÇÃO DE UM PROGRAMA DE CONTROLE DE FLUXO
FINANCEIRO**

O trabalho foi solicitado pelo professor de Algoritmos e Estrutura de Dados II, Edson Nascimento para obtenção de nota parcial por parte dos alunos no primeiro semestre de 2019.

MANAUS, AM
2019

Introdução

Em computação, uma árvore B é uma estrutura de dados em árvore, auto balanceada, que armazena dados classificados e permite pesquisas, acesso sequencial, inserções e remoções em tempo logarítmico.

É normalmente usada em bancos de dados e sistemas de arquivos e foi projetada para funcionar especialmente em memória secundária como um disco magnético ou outros dispositivos de armazenamento secundário. Muitos sistemas de bancos de dados usam árvores B ou variações da mesma para armazenar informações. Dentre suas propriedades ela permite a inserção, remoção e busca de chaves numa complexidade de tempo logarítmica e, por esse motivo, é muito empregada em aplicações que necessitam manipular grandes quantidades de informação tais como um banco de dados ou um sistema de arquivos.

Árvores B são uma generalização das árvores binária de busca, pois cada nó de uma árvore binária armazena uma única chave de busca, enquanto as árvores B armazenam um número maior do que um de chaves de busca em cada nó, ou no termo mais usual para essa árvore, em cada página. Como a ideia principal das árvores B é trabalhar com dispositivos de memória secundária, quanto menos acessos a disco a estrutura de dados proporcionar, melhor será o desempenho do sistema na operação de busca sobre os dados manipulados.

Inventada por Rudolf Bayer e Edward Meyers McCreight em 1971 enquanto trabalhavam no Boeing Scientific Research Labs, a origem do nome (árvore B) não foi definida por estes. Especula-se que o B venha da palavra balanceamento, do nome de um de seus inventores Bayer ou de Boeing, nome da empresa.

Implementação

- O programa foi construído para um pequeno empreendedor, com o objetivo de controlar o fluxo financeiro de uma loja, afim de facilitar a administração.
- Este pode fazer lançamentos de crédito/débito, apresentar relatórios diários, mensais ou anuais mostrando os valores de entrada/saída e lucros/prejuízos.
- Todo o sistema utiliza de uma árvore-B para armazenar os dados. Com a possibilidade de fazer inserção, remoção na mesma.
- O programa é baseado no uso de chaves, que são geradas na hora da inserção do lançamento.
- A chave gerada é um número de 9 dígitos, sendo esta sua composição.
- 1º e 2º dígitos: Dia do lançamento.
- 3º e 4º dígitos: Mês do lançamento.
- 5º e 6º dígitos: Ano do lançamento.
- 7º, 8º e 9º dígitos: Número do lançamento no dia.
- Ex: 060219023
- Para o código acima, o algoritmo entende que este é o 23º lançamento do dia 06/02/2019
- O código é usado tanto para consulta de dados, quanto para remoção. Por isso é imprescindível que o usuário saiba fazer o uso corretamente desta informação.

Execução do Algoritmo

O programa, ao ser executado, está encarregado de imprimir na tela um menu, que por sua vez, irá mostrar as opções disponíveis ao usuário.

```
-----  
--                               --  
--           Arvore-B           --  
--       Desenvolvido por:       --  
--   Jean Cleison Braga Guimaraes - 21601227   --  
-----  
  
1) Nova Arvore  
2) Sobre  
3) Sair  
  
Digite uma opcao:
```

A seguir há a explicação de cada opção:

1 - Nova Arvore:

- Inicia o programa com a construção de uma árvore de ordem informada pelo usuário.

Tela da execução:

```
-----  
--                               --  
--           Arvore-B           --  
--       Desenvolvido por:       --  
--   Jean Cleison Braga Guimaraes - 21601227   --  
-----  
  
Um no de uma Arvore-B de ordem T deve ter no minimo T-1  
e no maximo 2*T filhos  
  
Digite a ordem T da sua B-Tree:
```

```
-----
--                               Arvore-B                               --
--                               Desenvolvido por:                       --
--                               Jean Cleison Braga Guimaraes - 21601227  --
-----

B-Tree de ordem 5 alocada.

1) Lancamentos
2) Consultas
3) Balanco Geral
4) Arvore
5) Excluir Arvore

Digite uma opcao:
```

1.1 - Lançamentos:

- Para lançamento e remoção de créditos e débitos.

Tela de execução:

```
1 - Inserir
2 - Remover

Digite uma opcao:
```

1.1.1 - Inserir:

- Opção utilizada para realizar um lançamento de crédito ou débito.

Tela de execução:

```
0 - Credito
1 - Debito

Digite uma opcao: 0

Digite a data(DD/MM/AAAA):13/06/2019

Valor:580

1-Dinheiro
2-Cheque
3-Boleto
4-Cartao
Forma de pagamento:2
Descricao da entrada:VENDA RELOGIO
```

1.1.2 - Remover:

- Opção utilizada para remover um lançamento.
- É necessário ter o código(chave) do lançamento.

Tela de execução:

```
Digite uma opcao: 2

Digite uma CHAVE para remover: 130619001
```

1.2 - Consultas:

- Para fazer consultas de um código (chave) ou mostrar relatórios diários, mensais, anuais.
- Informa os valores creditados e debitados.

Tela de execução:

```
1 - Por codigo
2 - Por dia
3 - Por mes
4 - Por ano

Digite uma opcao: 3

Digite o mes desejado(MM/AAAA):05/2019
Cod: 010519001.....Entrada.....01/05/2019.....Valor: R$3800.00.....Pagamento no Cartao.....Valor da parcela: R$633.33.....Parcela atual: 4/6.....VENDA PC GAMER
Cod: 020519001.....Entrada.....02/05/2019.....Valor: R$900.00.....Pagamento no Cartao.....Valor da parcela: R$300.00.....Parcela atual: 3/3.....VENDA NOTEBOOK
Cod: 230519001.....Entrada.....23/05/2019.....Valor: R$600.00.....Pagamento no Dinheiro.....VENDA MONITOR
Cod: 250519001.....Saida.....25/05/2019.....Valor: R$-53.40.....Pagamento no Boleto.....CONTA AGUA

Valor Creditado: R$1533.33
Valor Debitado: R$-53.40
Total: R$1479.93
Pressione ENTER para continuar.
```

1.3 - Balanço Geral:

- Mostra todos os lançamentos.

Tela de execução:

```
Digite uma opcao: 3

Cod: 010219001.....Entrada.....01/02/2019.....Valor: R$3800.00.....Pagamento no Cartao.....Valor da parcela: R$633.33.....Parcela atual: 1/6.....VENDA PC GAMER
Cod: 010319001.....Entrada.....01/03/2019.....Valor: R$3800.00.....Pagamento no Cartao.....Valor da parcela: R$633.33.....Parcela atual: 2/6.....VENDA PC GAMER
Cod: 010419001.....Entrada.....01/04/2019.....Valor: R$3800.00.....Pagamento no Cartao.....Valor da parcela: R$633.33.....Parcela atual: 3/6.....VENDA PC GAMER
Cod: 010519001.....Entrada.....01/05/2019.....Valor: R$3800.00.....Pagamento no Cartao.....Valor da parcela: R$633.33.....Parcela atual: 4/6.....VENDA PC GAMER
Cod: 010619001.....Entrada.....01/06/2019.....Valor: R$3800.00.....Pagamento no Cartao.....Valor da parcela: R$633.33.....Parcela atual: 5/6.....VENDA PC GAMER
Cod: 010719001.....Entrada.....01/07/2019.....Valor: R$3800.00.....Pagamento no Cartao.....Valor da parcela: R$633.33.....Parcela atual: 6/6.....VENDA PC GAMER
Cod: 020319001.....Entrada.....02/03/2019.....Valor: R$900.00.....Pagamento no Cartao.....Valor da parcela: R$300.00.....Parcela atual: 1/3.....VENDA NOTEBOOK
Cod: 020419001.....Entrada.....02/04/2019.....Valor: R$900.00.....Pagamento no Cartao.....Valor da parcela: R$300.00.....Parcela atual: 2/3.....VENDA NOTEBOOK
Cod: 020519001.....Entrada.....02/05/2019.....Valor: R$900.00.....Pagamento no Cartao.....Valor da parcela: R$300.00.....Parcela atual: 3/3.....VENDA NOTEBOOK
Cod: 020619001.....Entrada.....02/06/2019.....Valor: R$130.00.....Pagamento no Cartao.....Valor da parcela: R$130.00.....Parcela atual: 1/1.....SERVICO CONTABIL
Cod: 230519001.....Entrada.....23/05/2019.....Valor: R$600.00.....Pagamento no Dinheiro.....VENDA MONITOR
Cod: 250519001.....Saida.....25/05/2019.....Valor: R$-53.40.....Pagamento no Boleto.....CONTA AGUA
Cod: 300419001.....Saida.....30/04/2019.....Valor: R$-253.45.....Pagamento no Boleto.....CONTA DE LUZ
Cod: 300419002.....Entrada.....30/04/2019.....Valor: R$997.00.....Pagamento no Cheque.....PAGAMENTO
Cod: 010619001.....Saida.....01/06/2019.....Valor: R$-560.00.....Pagamento no Boleto.....FATURA CARTAO
Cod: 060619001.....Entrada.....06/06/2019.....Valor: R$354.00.....Pagamento no Dinheiro.....SERVICO TI

Valor Creditado: R$6781.00
Valor Debitado: R$-866.85
Total: R$5914.15
```

1.4 - Arvore:

- Mostra os nós e as chaves da Árvore-B.

Tela de execução:

```
Digite uma opcao: 4

Mostrando a arvore:
  Algum no no nivel 1 possui 5 chave(s): 10219001 10319001 10419001 10519001 10619001
  Algum no no nivel 1 possui 5 chave(s): 10719001 20319001 20419001 20519001 20619001
  Algum no no nivel 1 possui 4 chave(s): 230519001 250519001 300419001 300419002
  Algum no no nivel 0 possui 2 chave(s): 10619001 60619001
Pressione ENTER para continuar.
```

2 - Sobre:

- Mostra uma pequena informação sobre o programa.

Tela de execução:

```
-----
--                               --
--               Arvore-B        --
--           Desenvolvido por:    --
--   Jean Cleison Braga Guimaraes - 21601227  --
--                               --
-----
--                               --
--   Trabalho apresentado na disciplina AED2    --
--   ministrada pelo Prof. Edson                --
--   UFAM                                       --
--                               --
-----

Pressione ENTER para voltar ao menu.
```

3 - Sair:

- Finaliza o programa.

Estrutura do Algoritmo

Além dos algoritmos relativos à árvore B, como os de busca, inserção, impressão, rotação, foram criadas estruturas e algoritmos afim de aperfeiçoar o sistema. Segue abaixo as principais estruturas e funções utilizadas.

Principais estruturas

pair <int key, void* value>

Como o valor associado à chave é um ponteiro void, a Árvore-B terá capacidade de armazenar qualquer tipo de dado indexável por uma chave inteira.

```
typedef struct pair_t {  
    int key;  
    void *value;  
} pair_t;
```

Nó da Árvore-B.

is_leaf: esse é um nó-folha?

n_keys: número de chaves ativas.

children: ponteiro para a lista de (ponteiros para) nós-filhos.

keys: ponteiro para a lista de (ponteiros para) pairs<key, value>.

```
typedef struct node_t {  
    bool is_leaf;  
    int n_keys;  
    struct node_t **children;  
    pair_t **keys;  
} node_t;
```

Posição de uma chave/elemento da Árvore-B.

Atenção ao uso:

Expõe acesso direto à um nó da Árvore-B, incluindo suas chaves e nós-filhos.

node: ponteiro para o nó.

index: índice da chave dentro do nó.

```
typedef struct node_position_t {  
    node_t *node;  
    int index;  
} node_position;
```

Estrutura da Árvore.

order: número da ordem da Arvore-B (cada nó da Arvore-B possui [order-1, 2*order] filhos).

root: ponteiro para o nó raiz da Arvore-B (possui [2, 2*order] filhos).

```
typedef struct btree_t {  
    int order;  
    node_t *root;  
} BTree;
```

Estrutura dos dados.

```
typedef struct entrada{  
    TData data; //dia, mês, ano  
    int tipo;  
    int forma_pagamento;  
    double valor;  
    double valor_parcela;  
    int n_parcelas;  
    int parcela_atual;  
    char descr[50];  
    int chave;  
}TEntada;
```

Estrutura para guardar valores creditados e debitados.

```
typedef struct financas{  
    double valor_entrada;  
    double valor_saida;  
}TFinancas;
```

Principais Funções

Função de leitura de entrada (Credito ou Debito)

```
TEntrada* lerEntrada(int tipo,int dia,int mes,int ano,double valor,int formapagamento,int nparcelas,double valorparcela,int parcelaatual,char descri[50]){
    TEntrada *ex = malloc(sizeof(TEntrada));
    ex->tipo = tipo;
    ex->data.dia = dia;
    ex->data.mes = mes;
    ex->data.ano = ano;
    ex->valor = valor;
    if(ex->tipo==1)
        ex->valor = -valor;
    ex->forma_pagamento = formapagamento;
    ex->n_parcelas = nparcelas;
    ex->valor_parcela = (valor/nparcelas);
    if(ex->tipo==1)
        ex->valor_parcela = -(ex->valor_parcela);
    ex->parcela_atual = parcelaatual;

    strcpy((ex->descr),descri);
    int data = (((dia*100)+mes)*100)+(ano%100);
    int chave1 = (data*1000)+1;
    ex->chave = chave1;
    return ex;
};
```

Função que insere os valores na arvore

```
BTree* inserirEntrada(BTree *arvore){
    int tipo;
    printf("0 - Credito\n1 - Debito\n");
    printf("\n");
    printf(" Digite uma opcao: ");
    scanf("%d",&tipo);
    int dia,mes,ano;
    printf("\nDigite a data(DD/MM/AAAA):");
    scanf("%d/%d/%d",&dia,&mes,&ano);
    double valor;
    printf("\nValor:");
    scanf("%lf",&valor);
    int formapagamento;
    printf("\n1-Dinheiro\n2-Cheque\n3-Boleto\n4-Cartao\nForma de
pagamento:");
    scanf("%d",&formapagamento);
    int nparcelas=1;
    if(formapagamento==4){
        printf("1 - A Vista\n2 - 2x\n3 - 3x\n4 - 4x\n5 - 5x\n6
- 6x\nParcelado em:");
        scanf("%d",&nparcelas);
    }
    double valorparcela = valor/nparcelas;
    int parcelaatual=1;
    char descri[50];
    getchar();
    printf("Descricao da entrada:");
    fgets(descri,100,stdin);
    int i;
    TEntrada *entrada;
    for(i=0;i<nparcelas;i++){
        entrada = lerEntrada(tipo,dia,mes,ano,valor,formapagam
ento,nparcelas,valorparcela,parcelaatual,descri);
```

```

        mes++;
        if(mes>12){
            mes=1;
            ano++;
        }
        parcelaatual++;
        node_position info = btree_insert(arvore, (entrada->ch
ave), entrada);
        while(info.node == NULL){
            (entrada->chave)++;
            info = btree_insert(arvore, (entrada->ch
ave), entrada);
        }
    }
    return arvore;
}

```

Função que mostra o relatório anual de lançamentos

```

void listar_nos_ano(node_t *node, int level, int a) {
    assert(node != NULL);
    int i;

    if (!node->is_leaf) {
        for (i = 0; i < node->n_keys+1; ++i) {
            listar_nos_ano(node->children[i], level+1,a);
        }
    }
    TEntrada *b;
    for (i = 0; i < node->n_keys; ++i) {
        b = (node->keys[i]->value);
        if((b->data.ano)==a){

            printf("Cod: %.9d.....", node->keys[i]->key);
            if(!(b->tipo)){

```

```

        printf("Entrada.....");
        tf->valor_entrada = tf->valor_entrada + b->valor_entrada;
    }
    else {
        printf("Saida.....");
        tf->valor_saida = tf->valor_saida + b->valor_saida;
    }
    printf("%.2d/%.2d/%d.....", (b->data.dia), (b->data.mes), (b->data.ano));
    printf("Valor: R$%.2lf.....", (b->valor));
    printf("Pagamento no ");
    switch(b->forma_pagamento){
        case 1: printf("Dinheiro.....");
                break;
        case 2: printf("Cheque.....");
                break;
        case 3: printf("Boleto.....");
                break;
        case 4: printf("Cartao.....");
                break;
    }
    if(b->forma_pagamento==4){
        printf("Valor da parcela: R$%.2lf.....", (b->valor_parcela));
        printf("Parcela atual: %d/%d.....", (b->parcela_atual), (b->n_parcelas));
    }
    printf("%s", (b->descr));
}
}
}
}

```

Corpo da Função main()

```
1300
1301 int main() {
1302     screen next_screen = HOME;
1303     snprintf(msg, MSG_LEN, "-");
1304     n_keys = 0;
1305
1306     while (next_screen != EXIT) {
1307         switch (next_screen) {
1308             case HOME:
1309                 next_screen = home_screen();
1310                 break;
1311             case ABOUT:
1312                 next_screen = about_screen();
1313                 break;
1314             case RUN:
1315                 next_screen = run_screen();
1316                 break;
1317             case RUNNING:
1318                 next_screen = running_screen();
1319                 break;
1320             case EXIT:
1321                 break;
1322         }
1323     }
1324
1325     snprintf(msg, MSG_LEN, " Saindo da aplicacao");
1326     goodbye();
1327
1328     return 0;
1329 }
1330
```

Referências Bibliográficas

1. https://pt.wikipedia.org/wiki/%C3%81rvore_B
2. <https://www.ime.usp.br/~pf/algoritmos/aulas/string.html>
3. <https://www.unicamp.br/fea/ortega/info/aula09.htm>
4. <https://pt.stackoverflow.com/questions/254652/qual-%C3%A9-a-fun%C3%A7%C3%A3o-do-getch-e-da-biblioteca-conio-h>
5. <https://www.cprogressivo.net/2013/03/Lendo-e-Escrevendo-Strings-em-C.html>
6. <http://linguagemc.com.br/string-em-c-vetor-de-caracteres/>
7. <https://stackoverflow.com/questions/3025050/error-initializer-element-is-not-constant-when-trying-to-initialize-variable-w>
8. <https://github.com/AnthonyBobsin/2-3Binary-Tree/blob/master/btree.c>
9. <https://www.geeksforgeeks.org/b-tree-set-1-introduction-2/>
10. <https://www.ime.usp.br/~pf/estruturas-de-dados/aulas/B-trees.html>
11. <https://github.com/falcaopetri/B-Tree>