

## 2ª Trabalho

Curso: Engenharia da Computação  
Disciplina: Estruturas de Dados  
Prof. Jarbas Joaci de Mesquita Sá Junior  
Universidade Federal do Ceará – UFC/Sobral

**Entrega:** 12/03/2021 via e-mail para jarbas\_joaci@yahoo.com.br **Obs.:** 1ª - O trabalho é individual e não será recebido após a data mencionada. 2ª - Preferencialmente fazer o trabalho usando a IDE Dev-C++. Enviar todos os arquivos do projeto, exceto os executáveis (.exe).

1. Implemente a TAD “arvb.h” (Árvore Binária de Buscas) e acrescente as seguintes funções:

a) função que retorne a quantidade de folhas de uma árvore binária de busca que possuem no campo `info` um número divisível por `n`. Essa função deve obedecer ao protótipo (2,0 pontos):

```
int folhas_div_n(ArvB* a, int n);
```

b) função que decida se uma árvore é degenerada. Essa função deve obedecer ao protótipo (2,0 pontos):

```
int isdegenerada(ArvB* a);
```

Obs. 1 – verdadeiro, 0 – falso.

c) função que, dada uma árvore binária de busca, retorne a quantidade de nós que tenham fator de balanceamento igual a `n`. Essa função deve obedecer ao protótipo (2,0 pontos):

```
int nos_fb_n(ArvB* a, int n);
```

d) função que imprima os elementos em comum de duas árvores binárias de busca. Essa função deve obedecer ao protótipo (2,0 pontos):

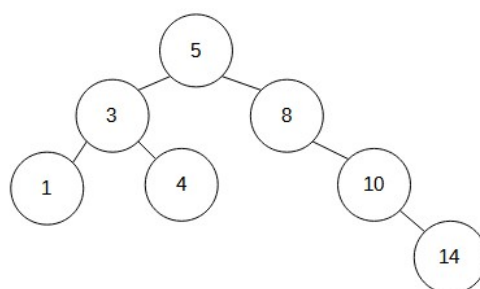
```
void impressao_arv_elem_comuns(ArvB* a, ArvB* b);
```

e) função que imprima os elementos de uma árvore binária de busca por níveis. Essa função deve obedecer ao protótipo (2,0 pontos):

```
void impressao_arv_niveis(ArvB* a);
```

Por exemplo, na árvore da figura abaixo, a impressão deve ser:

5 – nível 0  
3, 8 – nível 1  
1, 4, 10 – nível 2  
14 – nível 3



A seguir, execute o seguinte programa.

```
#include <stdio.h>
#include <stdlib.h>
#include "arvb.h"

int main(void){

    ArvB* arv1 = arvb_cria_vazia();

    arv1=arvb_insere(arv1,41);
    arv1=arvb_insere(arv1,7);
    arv1=arvb_insere(arv1,55);
    arv1=arvb_insere(arv1,71);
    arv1=arvb_insere(arv1,40);
    arv1=arvb_insere(arv1,10);
    arv1=arvb_insere(arv1,6);
    arv1=arvb_insere(arv1,66);

    printf("Qtd folhas div por 3 %d\n",folhas_div_n(arv1,3));
    printf("Arv1 é degenerada? %d\n",isdegenerada(arv1));
    printf("Nós com FBs iguais a 1 %d\n",nos_fb_n(arv1,1));
    printf("Nós com FBs iguais a 2 %d\n",nos_fb_n(arv1,2));

    ArvB* arv2 = arvb_cria_vazia();

    arv2=arvb_insere(arv2,7);
    arv2=arvb_insere(arv2,8);
    arv2=arvb_insere(arv2,9);
    arv2=arvb_insere(arv2,11);
    arv2=arvb_insere(arv2,71);

    printf("Arv2 é degenerada? %d\n",isdegenerada(arv2));

    impressao_arv_elem_comuns(arv1,arv2);

    impressao_arv_niveis(arv1);

    arvb_libera(arv1);
    arvb_libera(arv2);

    system("PAUSE");
    return 0;

}
```