



Instituto Federal de Educação, Ciência e Tecnologia de Brasília – Câmpus Taguatinga
Ciência da Computação – Estrutura de Dados e Algoritmos
Prova III – 1º/2017 – Árvores
Prof. Daniel Saad Nogueira Nunes

Aluno: _____

Matrícula: _____

Data: 5 de Julho de 2017

Duração da prova: 100 minutos

Tabela de notas (uso exclusivo do professor)

Questão	Pontos	Nota
1	3	
2	3	
3	3	
4	3	
Total	12	

Observações

- Esta prova tem o total de 3 páginas (incluindo a capa) e 4 questões.
- O número total de pontos é 12.
- Certifique-se de assinar todas as folhas de resposta bem como a capa da prova.
- Leia atentamente todas as questões da prova. A interpretação do problema é crucial para o desenvolvimento correto da resposta.
- Resoluções sem justificativa não serão consideradas.
- É vedado o uso de equipamentos eletrônicos, como celulares, notebooks entre outros.
- A prova será **anulada** e medidas disciplinares serão tomadas para os alunos que “colarem” durante a avaliação.

★ Certifique-se de assinar todas as folhas de resposta.

Questão 1 (3 pontos)

Dada uma árvore binária de pesquisa, elabore uma função que determina se um dado encontra-se ou não na árvore. Ele deve possuir a seguinte assinatura:

```
int bst_find(bst_t* t, void* data);
```

Questão 2 (3 pontos)

Dada uma árvore binária de pesquisa, elabore um algoritmo para remover um dado elemento `data` desta árvore. Ele deve possuir a seguinte assinatura:

```
void bst_remove(bst_t* t, void* data);
```

Questão 3 (3 pontos)

Seja T uma árvore AVL vazia. Informe e ilustre as sequências de operações feitas para deixá-la balanceada mediante a inserção dos elementos $(10, 77, 75, 7, 5, 3, 4)$ nesta ordem.

Questão 4 (3 pontos)

De acordo com as rotações em árvores AVL:

- (a) (1.5 pontos) Implemente uma rotação para a esquerda em uma subárvore com raiz em x . Sua função deverá possuir a seguinte assinatura:

```
avl_node_t* avl_left_rotate(avl_node_t* x);
```

- (b) (1.5 pontos) Implemente uma rotação para a direita em uma subárvore com raiz em y . Sua função deverá possuir a seguinte assinatura:

```
avl_node_t* avl_right_rotate(avl_node_t* y)
```

```
typedef struct bst_node_t{
    void* data;
    struct bst_node_t* left;
    struct bst_node_t* right;
}bst_node_t;

typedef struct bst_t{
    bst_node_t* root;
    bst_element_constructor_fn constructor;
    bst_element_destructor_fn destructor;
    bst_tree_element_compare_fn comparator;
    size_t size;
}bst_t;

typedef struct avl_node_t{
    void* data;
    size_t height;
    struct avl_node_t* left;
    struct avl_node_t* right;
}avl_node_t;

typedef struct avl_tree_t{
    struct avl_node_t* root;
    avl_tree_element_constructor_fn constructor;
    avl_tree_element_destructor_fn destructor;
    avl_tree_element_compare_fn comparator;
    size_t size;
}avl_tree_t;
```

Nadie nace sabiendo, todo lo
aprendemos con el tiempo

Fangio, J. M.