

Relatório de Documentação do Trabalho de AVL de CI 057 - Algoritmos e Estrutura de Dados III

Funções Públicas

nodo_t: Struct para os nodos da arvore. Guarda a altura, seu valor os ponteiros para seu pai, o nodo a sua esquerda e direita.

```
typedef struct nodo {  
    int value;  
    int height;  
    struct nodo *left, *right, *father;  
} nodo_t;
```

tree_t: Struct da arvore. Guarda um ponteiro para sua Raiz, e um inteiro que contem a quantidade de nodos na estrutura.

```
typedef struct tree {  
    nodo_t *root;  
    int nodos_qtd;  
} tree_t;
```

tree_t *create_tree ()

Inicia os valores de uma arvore. Aloca espaço para suas propriedades e seta os valores iniciais entre 0 e NULL.

nodo_t *create_nodo (int value)

Inicia um Nodo e aloca espaço para suas propriedades. Recebe o valor do Nodo.

void in_order (tree_t tree, nodo_t nodo, int nivel)

Lê uma Arvore AVL usando o algoritmo tradicional (esquerda, raiz, direita) o retorno é formatado para exibir cada valor em uma linha, sendo o primeiro valor a chave do nodo e o segundo seu nível.

nodo_t search_tree (nodo_t nodo, int value)

Executa uma busca de um valor em uma arvore AVL, retorna NULL caso o valor não exista, ou o nodo no qual o valor é a chave, recebe o nodo raiz e o valor que será buscado.

nodo_t left_rotate_tree (tree_t tree, nodo_t *nodo, int d_value)

Rotaciona o nodo selecionado para a esquerda em relação ao seu filho da direita, O nodo a ser enviado é o pai do nodo o qual você deseja rotacionar. Atualiza as alturas caso necessário. A variável d_value indica se a rotação decrementará ou não valores nas alturas.

nodo_t right_rotate_tree (tree_t tree, nodo_t *nodo, int d_value)

O mesmo da função anterior, mas nesse caso para a direita.

nodo_t insert_in_leave (tree_t tree, nodo_t *nodo, int value)

Insere um valor nas folhas da árvore, após a inserção, verifica o balanceamento da árvore e a ajusta caso necessário.

nodo_t predecessor (nodo_t nodo)

Busca o antecessor do nodo indicado.

void nodo_transplant (tree_t tree, nodo_t removed, nodo_t *transplanted)

Realiza a troca do nodo 'removed' pelo nodo 'transplanted' por meio de um 'transplante', removendo os ponteiros que indicavam para o nodo 'removed'.

void transplant_nodo_delete (tree_t tree, nodo_t nodo)

Remove o nodo da árvore a partir de um 'transplante'. Executa as verificações de balanceamento após a remoção e realiza os ajustes necessários.

Funções Internas

int tree_height (nodo_t *nodo)

Retorna a altura do nodo indicado, caso seja NULL, retorna 0.

int get_balance (nodo_t *nodo)

Retorna a diferença da altura da subárvore da esquerda com a subárvore da direita.