My Project

Generated by Doxygen 1.8.11

Contents

Chapter 1

README

TRABALHO DA DISCIPLINA DE METODOS DE PROGRAMAÇÃO

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

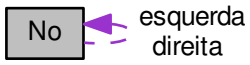
Here is a list of all documented files with brief descriptions:

Chapter 4

Class Documentation

4.1 No Struct Reference

Collaboration diagram for No:



Public Attributes

- int **numero**
- struct No * **esquerda**
- struct No * **direita**

The documentation for this struct was generated from the following file:

- arvoreBinaria.c

Chapter 5

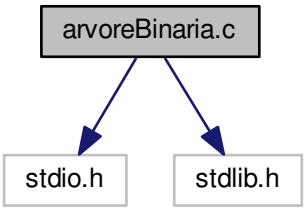
File Documentation

5.1 arvoreBinaria.c File Reference

Arquivo de teste da arvore binaria implementada em C.

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

Include dependency graph for arvoreBinaria.c:



Classes

- struct No

Typedefs

- typedef struct No **No**

Functions

- void criarArvore (No **pRaiz)
- void inserir (No **Raiz, int numero)
- No * **MaiorDireita** (No **no)
- No * **MenorEsquerda** (No **no)
- void **remover** (No **pRaiz, int numero)
- void **destruir** (No **pRaiz)
- void **exibirEmOrdem** (No *pRaiz)
- void **exibirPreOrdem** (No *pRaiz)
- void **exibirPosOrdem** (No *pRaiz)
- int **contarNos** (No *pRaiz)
- int **contarFolhas** (No *pRaiz)
- int **maior** (int a, int b)
- int **altura** (No *pRaiz)
- No * **busca_no** (No *raiz, int chave)
- void **busca** (No *raiz)

5.1.1 Detailed Description

Arquivo de teste da arvore binaria implementada em C.

Primeiro trabalho da disciplina Metodos de Programacao Aluno: Marcelo Axel Chiapinotto de Nazare

5.1.2 Function Documentation

5.1.2.1 criarArvore (No ** *pRaiz*)

Para criar a arvore criamos um novo ponteiro o qual ainda nao aponta para lugar nenhum.

5.1.2.2 inserir (No ** *Raiz,* int *numero*)

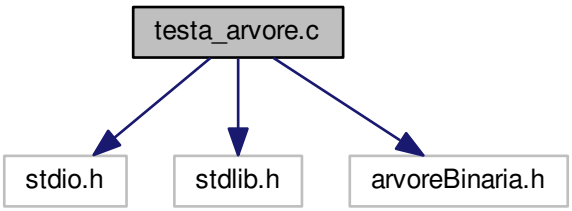
Para inserir um novo elemento na arvore biaria temos que alocar o espaço ncesssário para esse novo no e criar dois novos ponteiros que apontam para a folha da esquerda e da direita e um ponteiro para o conteudo desse novo no. Teremos dois comportamentos para colocar a arvore em ordem. Se a raiz existir mas estiver vazia apenas vamos inserir, caso contrario iremos verificar recursivamente o local para que os nos fiquem em ordem crescente.

5.2 testa_arvore.c File Reference

Arquivo de teste da arvore binaria implementada em C.

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

Include dependency graph for testa_arvore.c:



Functions

- int **main** ()

5.2.1 Detailed Description

Arquivo de teste da arvore binaria implementada em C.

Primeiro trabalho da disciplina Metodos de Programacao Aluno: Marcelo Axel Chiapinotto de Nazare

Index