Vinte Perguntas

Generated by Doxygen 1.8.11

Contents

1 Class Index **1**

1.1 Class List 1

2 File Index **3**

2.1 File List 3

3 Class Documentation **5**

3.1 node Struct Reference 5

3.1.1 Member Data Documentation 5

3.1.1.1 nao 5

3.1.1.2 pergunta 5

3.1.1.3 sim 5

3.1.1.4 valor 5

4 File Documentation **7**

4.1 arvore.txt File Reference 7

4.2 funcoes.c File Reference 7

4.2.1 Typedef Documentation 8

4.2.1.1 tree 8

4.2.2 Function Documentation 8

4.2.2.1 chooseMenu(tree *load) 8

4.2.2.2 clearScreen() 8

4.2.2.3 createGame(tree **atual, tree **anterior, int altura) 8

4.2.2.4 deleteTree(tree *del) 8

4.2.2.5 loadTree(tree **save, tree **anterior, FILE *fp, int posicao, int lado) 8

4.2.2.6 newNode(char *pergunta, int valor) 8

4.2.2.7 playGame(tree **atual, int altura) 8

4.2.2.8 saveTree(tree *save, FILE *fp) 8

4.3 main.c File Reference 8

4.3.1 Function Documentation 9

4.3.1.1 main() 9

Index **11**

Chapter 1

Class Index

1.1 Class List

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

node 5

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

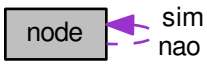
funcoes.c 7
main.c 8

Chapter 3

Class Documentation

3.1 node Struct Reference

Collaboration diagram for node:



Public Attributes

- int valor
- char * pergunta
- struct node * sim
- struct node * nao

3.1.1 Member Data Documentation

3.1.1.1 struct node* node::nao

3.1.1.2 char* node::pergunta

3.1.1.3 struct node* node::sim

3.1.1.4 int node::valor

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

- funcoes.c

Chapter 4

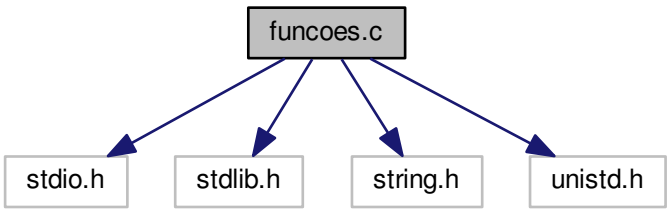
File Documentation

4.1 arvore.txt File Reference

4.2 funcoes.c File Reference

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

Include dependency graph for funcoes.c:



Classes

- struct node

Typedefs

- typedef struct node tree

Functions

- void clearScreen ()
- tree * newNode (char *pergunta, int valor)
- void deleteTree (tree *del)
- void saveTree (tree *save, FILE *fp)
- tree * loadTree (tree **save, tree **anterior, FILE *fp, int posicao, int lado)
- void createGame (tree **atual, tree **anterior, int altura)
- void playGame (tree **atual, int altura)
- void chooseMenu (tree *load)

4.2.1 Typedef Documentation

4.2.1.1 typedef struct node tree

4.2.2 Function Documentation

4.2.2.1 void chooseMenu (tree * *load*)

4.2.2.2 void clearScreen ( )

4.2.2.3 void createGame (tree ** *atual,* tree ** *anterior,* int *altura*)

4.2.2.4 void deleteTree (tree * *del*)

4.2.2.5 tree* loadTree (tree ** *save,* tree ** *anterior,* FILE * *fp,* int *posicao,* int *lado*)

4.2.2.6 tree* newNode (char * *pergunta,* int *valor*)

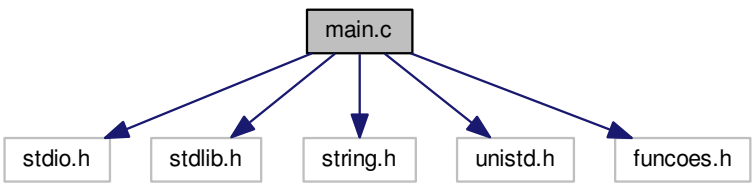
4.2.2.7 void playGame (tree ** *atual,* int *altura*)

4.2.2.8 void saveTree (tree * *save,* FILE * *fp*)

4.3 main.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <funcoes.h>
```

Include dependency graph for main.c:



Functions

- int main ()

4.3.1 Function Documentation

4.3.1.1 int main ( )

Index

arvore.txt, 7

chooseMenu
    funcoes.c, 8
clearScreen
    funcoes.c, 8
createGame
    funcoes.c, 8

deleteTree
    funcoes.c, 8

funcoes.c, 7
    chooseMenu, 8
    clearScreen, 8
    createGame, 8
    deleteTree, 8
    loadTree, 8
    newNode, 8
    playGame, 8
    saveTree, 8
    tree, 8

loadTree
    funcoes.c, 8

main
    main.c, 9
main.c, 8
    main, 9

nao
    node, 5
newNode
    funcoes.c, 8
node, 5
    nao, 5
    pergunta, 5
    sim, 5
    valor, 5

pergunta
    node, 5
playGame
    funcoes.c, 8

saveTree
    funcoes.c, 8
sim
    node, 5

tree
    funcoes.c, 8

valor
    node, 5