

Reference Manual

Generated by Doxygen 1.6.1

Sat Apr 3 21:46:23 2010

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	elemento Struct Reference	5
3.1.1	Detailed Description	5
3.1.2	Member Data Documentation	5
3.1.2.1	cor	5
3.1.2.2	letra	5
3.2	erro Struct Reference	6
3.2.1	Member Data Documentation	6
3.2.1.1	err_code	6
3.2.1.2	mensagem	6
4	File Documentation	7
4.1	ajuda.c File Reference	7
4.1.1	Detailed Description	7
4.1.2	Define Documentation	8
4.1.2.1	CHUNK_SIZE	8
4.1.3	Function Documentation	8
4.1.3.1	ajuda	8
4.1.3.2	E_Par	8
4.1.3.3	E_Par_des	8
4.1.3.4	pds	9
4.1.3.5	pis	9
4.1.3.6	snd	9

4.1.3.7	trp	10
4.1.3.8	vb	10
4.1.3.9	vl	10
4.1.3.10	vp	11
4.2	ajuda.h File Reference	12
4.2.1	Function Documentation	12
4.2.1.1	ajuda	12
4.2.1.2	E_Par	12
4.2.1.3	E_Par_des	12
4.2.1.4	pds	13
4.2.1.5	pis	13
4.2.1.6	snd	14
4.2.1.7	trp	14
4.2.1.8	vb	14
4.2.1.9	vl	15
4.2.1.10	vp	15
4.3	comandos.c File Reference	16
4.3.1	Detailed Description	16
4.3.2	Function Documentation	16
4.3.2.1	executa_comando	16
4.3.2.2	executaCriaJogo	17
4.3.2.3	executaGravaJogo	17
4.3.2.4	executaJogada	17
4.4	comandos.h File Reference	18
4.4.1	Function Documentation	18
4.4.1.1	executa_comando	18
4.4.1.2	executaCriaJogo	18
4.4.1.3	executaGravaJogo	18
4.4.1.4	executaJogada	19
4.5	erro.c File Reference	20
4.5.1	Typedef Documentation	20
4.5.1.1	ERRO	20
4.5.2	Function Documentation	20
4.5.2.1	mensagem_de_erro	20
4.6	erro.h File Reference	21
4.6.1	Define Documentation	21

4.6.1.1	E_ARGS	21
4.6.1.2	E_COMMAND	21
4.6.1.3	E_COORDS	21
4.6.1.4	E_FILE_FORMAT	21
4.6.1.5	E_INVARGS	21
4.6.1.6	E_NO_BOARD	21
4.6.1.7	E_NO_FILE	21
4.6.1.8	E_NO_MOVES	21
4.6.1.9	E_NOERR	21
4.6.1.10	E_SAVE	21
4.6.1.11	E_WRONG_SOLUTION	21
4.6.2	Function Documentation	21
4.6.2.1	mensagem_de_erro	21
4.7	main.c File Reference	23
4.7.1	Detailed Description	23
4.7.2	Function Documentation	23
4.7.2.1	main	23
4.8	tabuleiro.c File Reference	24
4.8.1	Detailed Description	24
4.8.2	Function Documentation	24
4.8.2.1	criarTabuleiro	24
4.8.2.2	dimensao	25
4.8.2.3	ficheiroValido	25
4.8.2.4	freeTab	25
4.8.2.5	gravaJogo	26
4.8.2.6	jogada	26
4.8.2.7	printTabuleiro	26
4.9	tabuleiro.h File Reference	27
4.9.1	Detailed Description	27
4.9.2	Typedef Documentation	27
4.9.2.1	Elem	27
4.9.2.2	Elemento	27
4.9.3	Function Documentation	27
4.9.3.1	criarTabuleiro	27
4.9.3.2	dimensao	28
4.9.3.3	ficheiroValido	28

4.9.3.4	<code>freeTab</code>	28
4.9.3.5	<code>gravaJogo</code>	29
4.9.3.6	<code>jogada</code>	29
4.9.3.7	<code>printTabuleiro</code>	29

Chapter 1

Class Index

1.1 Class List

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

elemento	(É constituída por dois chars, que correspondem às características de cada célula do tabuleiro, a cor e a letra)	5
erro	6

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

ajuda.c (Comandos de ajuda ao jogo)	7
ajuda.h	12
comandos.c (Comandos do jogo)	16
comandos.h	18
erro.c	20
erro.h	21
main.c (Main do jogo)	23
tabuleiro.c (Tabuleiro do jogo)	24
tabuleiro.h (Header file do tabuleiro.c)	27

Chapter 3

Class Documentation

3.1 elemento Struct Reference

É constituída por dois chars, que correspondem às características de cada célula do tabuleiro, a cor e a letra.

```
#include <tabuleiro.h>
```

Public Attributes

- char [letra](#)
Letra da célula do tabuleiro.
- char [cor](#)
Cor da célula do tabuleiro.

3.1.1 Detailed Description

É constituída por dois chars, que correspondem às características de cada célula do tabuleiro, a cor e a letra.

3.1.2 Member Data Documentation

3.1.2.1 elemento::cor

Cor da célula do tabuleiro.

3.1.2.2 elemento::letra

Letra da célula do tabuleiro.

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

- [tabuleiro.h](#)

3.2 erro Struct Reference

Public Attributes

- char * [err_code](#)
- char * [mensagem](#)

3.2.1 Member Data Documentation

3.2.1.1 char* erro::err_code

3.2.1.2 char* erro::mensagem

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

- [erro.c](#)

Chapter 4

File Documentation

4.1 ajuda.c File Reference

Comandos de ajuda ao jogo. #include "tabuleiro.h"

```
#include "erro.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <fcntl.h>
```

Defines

- #define [CHUNK_SIZE](#) 32

Functions

- int [ajuda](#) ()
- int [trp](#) ([Elem](#) **Tab, int *DIM)
- int [snd](#) ([Elem](#) **Tab, int *DIM)
- int [E_Par](#) ([Elem](#) **Tab, int pos, char letra, int *DIM, char *local)
- int [pis](#) ([Elem](#) **Tab, int *DIM)
- int [vb](#) ([Elem](#) **Tab, int *DIM)
- int [vp](#) ([Elem](#) **Tab, int *DIM)
- int [vl](#) ([Elem](#) **Tab, int *DIM)
- int [E_Par_des](#) ([Elem](#) **Tab, char letra1, char letra2, int *DIM, int line1, int cl1, int line2, int cl2)
- int [pds](#) ([Elem](#) **Tab, int *DIM)

4.1.1 Detailed Description

Comandos de ajuda ao jogo.

Author:

André Pimenta, João Gomes, Nelson Carvalho

Date:

3 Abril de 2010

4.1.2 Define Documentation**4.1.2.1 #define CHUNK_SIZE 32****4.1.3 Function Documentation****4.1.3.1 int ajuda ()**

Função que vai imprimir as intruções de ajuda guardada no ficheiro com as instruções de ajuda

4.1.3.2 int E_Par (Elem ** *Tab*, int *pos*, char *letra*, int * *DIM*, char * *local*)

Função auxiliar que verifica se existem pares de letras numa linha ou numa coluna Para tal vai percorrer a linha ou coluna e verifica se existe. A função tem em conta o caso de tres letras seguidas não ser um par mas sim um triplo.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

pos Linha ou coluna onde vai procurar os pares

letra a letra que se pretende saber se existe pares ou não

local Para decidir se se procura nas linhas ou nas colunas

Returns:

Inteiro que confirma se existe ou não um para na coluna/linha.

4.1.3.3 int E_Par_des (Elem ** *Tab*, char *letra1*, char *letra2*, int * *DIM*, int *line1*, int *cl1*, int *line2*, int *cl2*)

Função auxiliar que verifica se existem pares de letras tanto nas linhas como nas colunas passdas como parametro. Para tal vai percorrer a linha ou coluna e verifica se existe.

Parameters:

Tab Tabuleiro do jogo

letra1 Primeira letra da diagonal.

letra2 Segunda letra da diagonal

DIM dimensão do tabuleiro

line1 Linha da primeira letra

line2 Linha da segunda letra

cl1 Coluna da primeira letra

cl2 Coluna da segunda letra

Returns:

Inteiro que confirma se existe ou não um para na coluna/linha.

4.1.3.4 int pds (Elem ** *Tab*, int * *DIM*)

Função responsável por ver se existem ou não pares isolados no tabuleiro. Para tal esta função vai pegar em quatro células, formando um quadrado. Neste quadrado verifica se as duas diagonais pertencem ou não a regra pares desencontrados, com a ajuda da função auxiliar EpParDes. Caso pertence ve se a diagonal contrária tem cores contrárias, se tiver dá uma mensagem de [erro](#), se não muda as cores.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#), é exibido através da função mensagem de [erro](#).

4.1.3.5 int pis (Elem ** *Tab*, int * *DIM*)

Função que procura pares isolados no tabuleiro (numa coluna ou numa linha). Para tal vai se percorrer todas as posições do tabuleiro, e ver se a letra é diferente das que a rodeiam na horizontal ou na vertical. Confirmando se a presença de uma letra isola, chama a função auxiliar Epar que vai ver se existe um par dessa letra na coluna ou na linha. Caso se verifiquem as duas condições irá ver se a letra não está de uma cor contrária à que se vai atribuir, caso não esteja atribui se uma nova cor, se não uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#), é exibido através da função mensagem de [erro](#).

4.1.3.6 int snd (Elem ** *Tab*, int * *DIM*)

Função que procura sandes no tabuleiro (três letras onde apenas a do meio é diferente na mesma linha/-coluna). Para tal vai se percorrer todas as posições do tabuleiro, ver caso existam quais letras de cima e de baixo, para verificar se existe triplo na vertical, e a da esquerda e a da direita para a horizontal. Confirmando se a presença de uma sandes de letras, atribui se as cores correctas caso nenhuma das células não tenha um cor contrária, se não envia uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#), é exibido através da função mensagem de [erro](#).

4.1.3.7 int trp (Elem ** *Tab*, int * *DIM*)

Função que procura triplos no tabuleiro (três letras iguais na mesma linha/coluna). Para tal vai se percorrer todas as posições do tabuleiro, ver caso existam quais letras de cima e de baixo, para verificar se existe triplo na vertical, e a da esquerda e a da direita para a horizontal. É também tido em conta o caso de existirem quadroplos não serem considerados triplos. Confirmando se a presença de um triplo atribui se as cores correctas caso nenhuma das células não tenha um cor contrária, se não envia uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#), é exibido através da função mensagem de [erro](#).

4.1.3.8 int vb (Elem ** *Tab*, int * *DIM*)

Função que percorre todas as posições do tabuleiro caso sejam pretas, verifica se não têm vizinas na vertical ou na horizontal com a cor preta. Caso existam retorna uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#), é exibido através da função mensagem de [erro](#).

4.1.3.9 int vl (Elem ** *Tab*, int * *DIM*)

Função que procura a existência de caminho entre todas as células brancas ou indefinidas. Caso não exista retorna uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#), é exibido através da função mensagem de [erro](#).

4.1.3.10 int vp (Elem ** *Tab*, int * *DIM*)

Função que percorre uma linha/coluna e ve se existem duas letras iguais com mesma cor(branca), fazendo o mesmo para todas as linhas/colunas. Caso existam retorna uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#), é exibido através da função mensagem de [erro](#).

4.2 ajuda.h File Reference

Functions

- int `ajuda` ()
- int `trp` (Elem **Tab, int *DIM)
- int `snd` (Elem **Tab, int *DIM)
- int `E_Par` (Elem **Tab, int pos, char letra, int *DIM, char *local)
- int `pis` (Elem **Tab, int *DIM)
- int `vb` (Elem **tab, int *DIM)
- int `vp` (Elem **tab, int *DIM)
- int `vl` (Elem **Tab, int *DIM)
- int `E_Par_des` (Elem **Tab, char letra1, char letra2, int *DIM, int line1, int cl1, int line2, int col2)
- int `pds` (Elem **Tab, int *DIM)

4.2.1 Function Documentation

4.2.1.1 int ajuda ()

Função que vai imprimir as intruções de ajuda guardada no ficheiro com as instruções de ajuda

4.2.1.2 int E_Par (Elem ** Tab, int pos, char letra, int * DIM, char * local)

Função auxiliar que verifica se existem pares de letras numa linha ou numa coluna Para tal vai percorrer a linha ou coluna e verifica se existe. A função tem em conta o caso de tres letras seguidas não ser um par mas sim um triplo.

Parameters:

Tab Tabuleiro do jogo
DIM dimensão do tabuleiro
pos Linha ou coluna onde vai procurar os pares
letra a letra que se pretende saber se existe pares ou não
local Para decidir se se procura nas linhas ou nas colunas

Returns:

Inteiro que confirma se existe ou não um para na coluna/linha.

4.2.1.3 int E_Par_des (Elem ** Tab, char letra1, char letra2, int * DIM, int line1, int cl1, int line2, int cl2)

Função auxiliar que verifica se existem pares de letras tanto nas linhas como nas colunas passdas como parametro. Para tal vai percorrer a linha ou coluna e verifica se existe.

Parameters:

Tab Tabuleiro do jogo
letra1 Primeira letra da diagonal.

letra2 Segunda letra da diagonal

DIM dimensão do tabuleiro

line1 Linha da primeira letra

line2 Linha da segunda letra

cl1 Coluna da primeira letra

cl2 Coluna da segunda letra

Returns:

Inteiro que confirma se existe ou não um para na coluna/linha.

4.2.1.4 int pds (Elem ** *Tab*, int * *DIM*)

Função responsável por ver se existem ou não pares isolados no tabuleiro. Para tal esta função vai pegar em quatro células, formando um quadrado. Neste quadrado verifica se as duas diagonais pertencem ou não a regra pares desencontrados, com a ajuda da função auxiliar EpParDes. Caso pertence ver se a diagonal contrária tem cores contrárias, se tiver dá uma mensagem de [erro](#), se não muda as cores.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#), é exibido através da função mensagem de [erro](#).

4.2.1.5 int pis (Elem ** *Tab*, int * *DIM*)

Função que procura pares isolados no tabuleiro (numa coluna ou numa linha). Para tal vai se percorrer todas as posições do tabuleiro, e ver se a letra é diferente das que a rodeiam na horizontal ou na vertical. Confirmando se a presença de uma letra isola, chama a função auxiliar Epar que vai ver se existe um par dessa letra na coluna ou na linha. Caso se verifiquem as duas condições irá ver se a letra não está de uma cor contrária à que se vai atribuir, caso não esteja atribui-se uma nova cor, se não uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#), é exibido através da função mensagem de [erro](#).

4.2.1.6 `int snd (Elem ** Tab, int * DIM)`

Função que procura sandes no tabuleiro(tres letras onde apenas a do meio é diferente na mesma linha/-coluna). Para tal vai se percorrer todas as posições do tabuleiro, ver caso existam quais letras de cima e de baixo, para verificar se existe triplo na vertical, e a da esquerda e a da direita para a horizontal. Confirmando se a presença de uma sandes de letras, atribui se as cores correctas caso nenhuma das celulas não tenha um cor contraria, se não envia uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#),é exibido atraves da função mensagem de [erro](#).

4.2.1.7 `int trp (Elem ** Tab, int * DIM)`

Função que procura triplos no tabuleiro(tres letras iguais na mesma linha/coluna). Para tal vai se percorrer todas as posições do tabuleiro, ver caso existam quais letras de cima e de baixo, para verificar se existe triplo na vertical, e a da esquerda e a da direita para a horizontal. É também tido em contam o caso de exitirem quadroplos não serem considerados triplos. Confirmando se a presença de um triplo atribui se as cores correctas caso nenhuma das celulas não tenha um cor contraria, se não envia uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#),é exibido atraves da função mensagem de [erro](#).

4.2.1.8 `int vb (Elem ** Tab, int * DIM)`

Função que percorre todas as posições do tabuleiro caso sejam pretas, verifica se não têm vizinas na vertical ou na horizontal com a cor preta. Caso existam retorna uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#),é exibido atraves da função mensagem de [erro](#).

4.2.1.9 int vl (Elem ** *Tab*, int * *DIM*)

Função que procura a existencia de caminho entre todas as celulas brancas ou indefinidas. Caso não exista retorna uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#),é exibido atraves da função mensagem de [erro](#).

4.2.1.10 int vp (Elem ** *Tab*, int * *DIM*)

Função que percorre uma linha/coluna e ve se existem duas letras iguais com mesma cor(branca), fazendo o mesmo para todas as linhas/colunas. Caso existam retorna uma mensagem de [erro](#).

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Inteiro que em caso de correr tudo bem retorna 0, caso exista um [erro](#),é exibido atraves da função mensagem de [erro](#).

4.3 comandos.c File Reference

Comandos do jogo. #include "tabuleiro.h"

#include "erro.h"

#include "ajuda.h"

#include <string.h>

#include <stdio.h>

#include <stdlib.h>

#include <ctype.h>

Functions

- int `executaJogada` (char *args, char *cor, int *DIM, Elem **Tab)
- int `executaCriaJogo` (char *args, int *DIM, Elem **Tab)
- int `executaGravaJogo` (char *args, int *DIM, Elem **Tab)
- int `executa_comando` (char *linha, int *DIM, Elem **Tab)

4.3.1 Detailed Description

Comandos do jogo.

Author:

André Pimenta, João Gomes, Nelson Carvalho

Date:

1 Abril de 2010

4.3.2 Function Documentation

4.3.2.1 int `executa_comando` (char * *linha*, int * *DIM*, Elem ** *Tab*)

A função `executa_comando` é uma das funções mais importantes pois é ela que vai reagir aos comandos dados pelo utilizador. A função irá receber a linha de comando separa la em dois, no comando e nos argumentos do comando, e passa los para as funções associadas aos comandos.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

linha Recebe uma string que corresponde á linha lida da shell.

Returns:

torna um inteiro que verifica se foi ou não bem sucedida a função.

4.3.2.2 int executaCriaJogo (char * *args*, int * *DIM*, Elem ** *Tab*)

A função executaCriaJog recebe os argumentos da linha de comando, vai verificar se tem os argumentos, certos se tal ocorre, executa criarJogo com o nome do argumento no formato ".ltr".

Parameters:

args Recebe uma string com os argumentos associados ao comando em questão ("cr").
Tab Tabuleiro do jogo
DIM dimensão do tabuleiro

Returns:

torna um inteiro que verifica se foi ou não bem sucedida a função.

4.3.2.3 int executaGravaJogo (char * *args*, int * *DIM*, Elem ** *Tab*)

A função executaGravaJogo recebe os argumentos da linha de comando e vai verificar se tem os argumetos certos, se tal for verdade, executa gravaJogo com o nome do argumento no formato ".ltr".

Parameters:

Tab Tabuleiro do jogo
DIM dimensão do tabuleiro
args Recebe uma string com os argumentos associados ao comando em questão("gr").

Returns:

torna um inteiro que verifica se foi ou não bem sucedida a função.

4.3.2.4 int executaJogada (char * *args*, char * *cor*, int * *DIM*, Elem ** *Tab*)

A função executajogada recebe os argumentos da linha de comandos e a cor que vai passar para a jogada. Vai verificar se tem os argumetos certos, e se tal vou verdade, chama a função jogada, para que esta se possa realizar.

Parameters:

args Recebe ums string com os argumentos do comando do utilizador.
cor Corresponde ao próprio comado que irá ser utilizado como a cor que se pretende mudar.
Tab Tabuleiro do jogo
DIM dimensão do tabuleiro

Returns:

Retorna um inteiro que verifica se foi ou não bem sucedida a função.

4.4 comandos.h File Reference

Functions

- int `executaJogada` (char *args, char *cor, int *DIM, Elem **Tab)
- int `executaCriaJogo` (char *args, int *DIM, Elem **Tab)
- int `executaGravaJogo` (char *args, int *DIM, Elem **Tab)
- int `executa_comando` (char *linha, int *DIM, Elem **Tab)

4.4.1 Function Documentation

4.4.1.1 int `executa_comando` (char * *linha*, int * *DIM*, Elem ** *Tab*)

A função `executa_comando` é uma das funções mais importantes pois é ela que vai reagir aos comandos dados pelo utilizador. A função irá receber a linha de comando separa la em dois, no comando e nos argumentos do comando, e passa los para as funções associadas aos comandos.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

linha Recebe uma string que corresponde á linha lida da shell.

Returns:

torna um inteiro que verifica se foi ou não bem sucedida a função.

4.4.1.2 int `executaCriaJogo` (char * *args*, int * *DIM*, Elem ** *Tab*)

A função `executaCriaJog` recebe os argumentos da linha de comando, vai verificar se tem os argumentos, certos se tal ocorre, executa `criarJogo` com o nome do argumento no formato ".ltr".

Parameters:

args Recebe uma string com os argumentos associados ao comando em questão ("cr").

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

torna um inteiro que verifica se foi ou não bem sucedida a função.

4.4.1.3 int `executaGravaJogo` (char * *args*, int * *DIM*, Elem ** *Tab*)

A função `executaGravaJogo` recebe os argumentos da linha de comando e vai verificar se tem os argumetos certos, se tal for verdade, executa `gravaJogo` com o nome do argumento no formato ".ltr".

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

args Recebe uma string com os argumentos associados ao comando em questão("gr").

Returns:

torna um inteiro que verifica se foi ou não bem sucedida a função.

4.4.1.4 int executaJogada (char * *args*, char * *cor*, int * *DIM*, Elem ** *Tab*)

A função executajogada recebe os argumentos da linha de comandos e a cor que vai passar para a jogada. Vai verificar se tem os argumetos certos, e se tal vou verdade, chama a função jogada, para que esta se possa realizar.

Parameters:

args Recebe ums string com os argumentos do comando do utilizador.

cor Corresponde ao próprio comado que irá ser utilizado como a cor que se pretende mudar.

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Returns:

Retorna um inteiro que verifica se foi ou não bem sucedida a função.

4.5 erro.c File Reference

```
#include <stdio.h>
#include "erro.h"
```

Classes

- struct [erro](#)

Typedefs

- typedef struct [erro](#) [ERRO](#)

Functions

- int [mensagem_de_erro](#) (int *erro_num*)

4.5.1 Typedef Documentation

4.5.1.1 typedef struct erro ERRO

4.5.2 Function Documentation

4.5.2.1 int mensagem_de_erro (int *erro_num*)

Reporta o [erro](#) escrevendo uma mensagem no standard error

Parameters:

erro_num O número do [erro](#) que vem da tabela de erros. Usar SEMPRE o define e não o valor numérico para maior legibilidade.

4.6 erro.h File Reference

Defines

- #define [E_NOERR](#) 0
- #define [E_COMMAND](#) 1
- #define [E_ARGS](#) 2
- #define [E_NO_BOARD](#) 3
- #define [E_NO_FILE](#) 4
- #define [E_FILE_FORMAT](#) 5
- #define [E_WRONG_SOLUTION](#) 6
- #define [E_NO_MOVES](#) 7
- #define [E_COORDS](#) 8
- #define [E_SAVE](#) 9
- #define [E_INVARGS](#) 10

Functions

- int [mensagem_de_erro](#) (int erro_num)

4.6.1 Define Documentation

4.6.1.1 #define E_ARGS 2

4.6.1.2 #define E_COMMAND 1

4.6.1.3 #define E_COORDS 8

4.6.1.4 #define E_FILE_FORMAT 5

4.6.1.5 #define E_INVARGS 10

4.6.1.6 #define E_NO_BOARD 3

4.6.1.7 #define E_NO_FILE 4

4.6.1.8 #define E_NO_MOVES 7

4.6.1.9 #define E_NOERR 0

4.6.1.10 #define E_SAVE 9

4.6.1.11 #define E_WRONG_SOLUTION 6

4.6.2 Function Documentation

4.6.2.1 int mensagem_de_erro (int *erro_num*)

Reporta o [erro](#) escrevendo uma mensagem no standard error

Parameters:

erro_num O número do [erro](#) que vem da tabela de erros. Usar SEMPRE o define e não o valor numérico para maior legibilidade.

4.7 main.c File Reference

```
main do jogo #include "tabuleiro.h"
#include "comandos.h"
#include "ajuda.h"
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <readline/readline.h>
#include <readline/history.h>
```

Functions

- int `main()`

4.7.1 Detailed Description

main do jogo

Author:

André Pimenta, João Gomes, Nelson Carvalho

Date:

5 Março de 2010

4.7.2 Function Documentation

4.7.2.1 int main()

A função main é um ciclo que apenas termina quando o utilizador insere o comando "q" Esta função vai então ser responsável por receber os comandos continuamente, e imprimir o resultado enquanto o jogo decorrer

4.8 tabuleiro.c File Reference

```
Tabuleiro do jogo. #include "tabuleiro.h"
#include "ajuda.h"
#include "erro.h"
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <readline/readline.h>
#include <readline/history.h>
```

Functions

- int [dimensao](#) (FILE *fp, int *DIM)
- int [ficheiroValido](#) (FILE *fp, int *DIM)
- void [freeTab](#) (int *DIM, Elem **Tab)
- int [criarTabuleiro](#) (char *nome, int *DIM, Elem **Tab)
- int [jogada](#) (char cor, int x, int y, int *DIM, Elem **Tab)
- void [printTabuleiro](#) (int *DIM, Elem **Tab)
- int [gravaJogo](#) (char *nome, int *DIM, Elem **Tab)

4.8.1 Detailed Description

Tabuleiro do jogo.

Author:

André Pimenta, João Gomes, Nelson Carvalho

Date:

5 Março de 2010

4.8.2 Function Documentation

4.8.2.1 int criarTabuleiro (char * *nome*, int * *DIM*, Elem ** *Tab*)

A função criarTabuleiro recebe o nome do ficheiro que queremos abrir e irá tentar abrir este. Depois irá chamar a função freeTab caso exista já um tabuleiro carregado e em seguida chama a função ficheiroValido para ver verificar se o mesmo é válido. Após todas verificações a função cria espaço para o tabuleiro na memória e carrega este para lá, ficando guardado no apontador Tab. Os valores lidos do ficheiro irão ser associados por duas características, a letra e a cor, esta ultima será "i" correspondente a indefinida caso a letra esteja rodeada por dois espaços brancos, preta caso esteja rodeada por dois parênteses rectos e branca caso esteja rodeada por dois parênteses curvos.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

nome Recebe uma string com o nome do ficheiro que vai abrir.

Returns:

Retorna um inteiro que verifica se conseguiu ou não criar o Tabuleiro.

4.8.2.2 int dimencao (FILE * *fp*, int * *DIM*)

A função *dimencao* é responsável por carregar a dimensão do tabuleiro que estamos a abrir, para a variável *DIM*. Para isso ela usa o *fscanf* da biblioteca *stdio*.

Parameters:

DIM dimensão do tabuleiro

fp Recebe o ficheiro do qual irá verificar se existe o valor da dim e passa-lo para a variável Global *DIM*.

Returns:

retorna um valor lógico 1 ou 0 que confirma se conseguiu ou não ler o valor correspondente á dimensão.

4.8.2.3 int ficheiroValido (FILE * *fp*, int * *DIM*)

A função *ficheiroValido* recebe o ficheiro do qual queremos importar o tabuleiro e verifica se este é válido para o nosso jogo. Para ser válido irá verificar se tem o número de elementos correspondentes a dimensão, se são elementos do tipo predefinido do tabuleiro(ex: "[A]")

Parameters:

DIM dimensão do tabuleiro

fp Recebe o ficheiro que está aberto na função *criarTabuleiro*

Returns:

Retorna um inteiro que irá confirmar se o ficheiro é válido ou não.

4.8.2.4 void freeTab (int * *DIM*, Elem ** *Tab*)

A função *freeTab* irá apagar da memória do tabuleiro antigo caso estejamos a abrir um novo tabuleiro.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Remarks:

Esta função apenas é usada se existir um tabuleiro carregado e queremos carregar outro.

4.8.2.5 int gravaJogo (char * *nome*, int * *DIM*, Elem ** *Tab*)

A função gravaJogo é responsável por gravar o nosso tabuleiro num ficheiro cujo o nome é dado como argumento e que está no formato ".ltr". O tabuleiro será imprimido para o ficheiro seguindo as regras de representação deste, para que possa ser lido de novo e seja considerado válido.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

nome Recebe uma string que representa o nome que vai dar ao ficheiro que vai guardar

4.8.2.6 int jogada (char *cor*, int *x*, int *y*, int * *DIM*, Elem ** *Tab*)

A função jogada é responsável por alterar os dados do tabuleiro. Ela irá receber o novo dado e as posições onde vai ficar o novo dado, actualizando assim o tabuleiro com os novos dados

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

cor nova cor para um elemento do tabuleiro

x Corresponde á linha que vamos substituir no tabuleiro.

y Corresponde á coluna que vamos substituir no tabuleiro.

Returns:

Retorna um inteiro que verifica se a jogada foi ou não bem sucedida.

4.8.2.7 void printTabuleiro (int * *DIM*, Elem ** *Tab*)

A função printTabuleiro é responsável por imprimir no ecrã o nosso tabuleiro.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

4.9 tabuleiro.h File Reference

header file do [tabuleiro.c](#) `#include <stdio.h>`

Classes

- struct [elemento](#)

É constituída por dois chars, que correspondem às características de cada célula do tabuleiro, a cor e a letra.

Typedefs

- typedef struct [elemento](#) [Elemento](#)
- typedef struct [elemento](#) * [Elem](#)

Functions

- int [dimensao](#) (FILE *fp, int *DIM)
- int [ficheiroValido](#) (FILE *fp, int *DIM)
- void [freeTab](#) (int *DIM, [Elem](#) **Tab)
- int [criarTabuleiro](#) (char *nome, int *DIM, [Elem](#) **Tab)
- int [jogada](#) (char cor, int x, int y, int *DIM, [Elem](#) **Tab)
- void [printTabuleiro](#) (int *DIM, [Elem](#) **Tab)
- int [gravaJogo](#) (char *nome, int *DIM, [Elem](#) **Tab)

4.9.1 Detailed Description

header file do [tabuleiro.c](#)

Author:

André Pimenta,Miguel Gomes,Nelson Cravalho

Date:

5 Março de 2010

4.9.2 Typedef Documentation

4.9.2.1 typedef struct elemento* Elem

4.9.2.2 typedef struct elemento Elemento

4.9.3 Function Documentation

4.9.3.1 int criarTabuleiro (char * nome, int * DIM, Elem ** Tab)

A função criarTabuleiro recebe o nome do ficheiro que queremos abrir e irá tentar abrir este. Depois irá chamar a função freeTab caso exista já um tabuleiro carregado e em seguida chama a função ficheiroValido

para ver verificar se o mesmo é válido. Após todas verificações a função cria espaço para o tabuleiro na memória e carrega este para lá, ficando guardado no apontador Tab. Os valores lidos do ficheiro irão ser associados por duas características, a letra e a cor, esta última será "i" correspondente a indefinida caso a letra esteja rodeada por dois espaços brancos, preta caso esteja rodeada por dois parênteses rectos e branca caso esteja rodeada por dois parênteses curvos.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

nome Recebe uma string com o nome do ficheiro que vai abrir.

Returns:

Retorna um inteiro que verifica se conseguiu ou não criar o Tabuleiro.

4.9.3.2 int dimensao (FILE * *fp*, int * *DIM*)

A função dimensao é responsável por carregar a dimensão do tabuleiro que estamos a abrir, para a variável DIM. Para isso ela usa o fscanf da biblioteca stdio.

Parameters:

DIM dimensão do tabuleiro

fp Recebe o ficheiro do qual irá verificar se existe o valor da dim e passa-lo para a variável Global DIM.

Returns:

retorna um valor lógico 1 ou 0 que confirma se conseguiu ou não ler o valor correspondente à dimensão.

4.9.3.3 int ficheiroValido (FILE * *fp*, int * *DIM*)

A função ficheiroValido recebe o ficheiro do qual queremos importar o tabuleiro e verifica se este é válido para o nosso jogo. Para ser válido irá verificar se tem o número de elementos correspondentes à dimensão, se são elementos do tipo predefinido do tabuleiro(ex: "[A]")

Parameters:

DIM dimensão do tabuleiro

fp Recebe o ficheiro que está aberto na função criarTabuleiro

Returns:

Retorna um inteiro que irá confirmar se o ficheiro é válido ou não.

4.9.3.4 void freeTab (int * *DIM*, Elem ** *Tab*)

A função freeTab irá apagar da memória do tabuleiro antigo caso estejamos a abrir um novo tabuleiro.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Remarks:

Esta função apenas é usada se existir um tabuleiro carregado e queremos carregar outro.

4.9.3.5 int gravaJogo (char * nome, int * DIM, Elem ** Tab)

A função gravaJogo é responsável por gravar o nosso tabuleiro num ficheiro cujo o nome é dado como argumento e que está no formato ".ltr". O tabuleiro será imprimido para o ficheiro seguindo as regras de representação deste, para que possa ser lido de novo e seja considerado válido.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

nome Recebe uma string que representa o nome que vai dar ao ficheiro que vai guardar

4.9.3.6 int jogada (char cor, int x, int y, int * DIM, Elem ** Tab)

A função jogada é responsável por alterar os dados do tabuleiro. Ela irá receber o novo dado e as posições onde vai ficar o novo dado, actualizando assim o tabuleiro com os novos dados

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

cor nova cor para um elemento do tabuleiro

x Corresponde á linha que vamos substituir no tabuleiro.

y Corresponde á coluna que vamos substituir no tabuleiro.

Returns:

Retorna um inteiro que verifica se a jogada foi ou não bem sucedida.

4.9.3.7 void printTabuleiro (int * DIM, Elem ** Tab)

A função printTabuleiro é responsável por imprimir no ecrã o nosso tabuleiro.

Parameters:

Tab Tabuleiro do jogo

DIM dimensão do tabuleiro

Index

- ajuda
 - ajuda.c, 8
 - ajuda.h, 12
- ajuda.c, 7
 - ajuda, 8
 - CHUNK_SIZE, 8
 - E_Par, 8
 - E_Par_des, 8
 - pds, 9
 - pis, 9
 - snd, 9
 - trp, 10
 - vb, 10
 - vl, 10
 - vp, 10
- ajuda.h, 12
 - ajuda, 12
 - E_Par, 12
 - E_Par_des, 12
 - pds, 13
 - pis, 13
 - snd, 13
 - trp, 14
 - vb, 14
 - vl, 14
 - vp, 15
- CHUNK_SIZE
 - ajuda.c, 8
- comandos.c, 16
 - executa_comando, 16
 - executaCriaJogo, 16
 - executaGravaJogo, 17
 - executaJogada, 17
- comandos.h, 18
 - executa_comando, 18
 - executaCriaJogo, 18
 - executaGravaJogo, 18
 - executaJogada, 19
- cor
 - elemento, 5
- criarTabuleiro
 - tabuleiro.c, 24
 - tabuleiro.h, 27
- dimencao
 - tabuleiro.c, 25
 - tabuleiro.h, 28
- E_ARGS
 - erro.h, 21
- E_COMMAND
 - erro.h, 21
- E_COORDS
 - erro.h, 21
- E_FILE_FORMAT
 - erro.h, 21
- E_INVARGS
 - erro.h, 21
- E_NO_BOARD
 - erro.h, 21
- E_NO_FILE
 - erro.h, 21
- E_NO_MOVES
 - erro.h, 21
- E_NOERR
 - erro.h, 21
- E_Par
 - ajuda.c, 8
 - ajuda.h, 12
- E_Par_des
 - ajuda.c, 8
 - ajuda.h, 12
- E_SAVE
 - erro.h, 21
- E_WRONG_SOLUTION
 - erro.h, 21
- Elem
 - tabuleiro.h, 27
- Elemento
 - tabuleiro.h, 27
- elemento, 5
 - cor, 5
 - letra, 5
- err_code
 - erro, 6
- ERRO
 - erro.c, 20
- erro, 6
 - err_code, 6

- mensagem, 6
- erro.c, 20
 - ERRO, 20
 - mensagem_de_erro, 20
- erro.h, 21
 - E_ARGS, 21
 - E_COMMAND, 21
 - E_COORDS, 21
 - E_FILE_FORMAT, 21
 - E_INVARGS, 21
 - E_NO_BOARD, 21
 - E_NO_FILE, 21
 - E_NO_MOVES, 21
 - E_NOERR, 21
 - E_SAVE, 21
 - E_WRONG_SOLUTION, 21
 - mensagem_de_erro, 21
- executa_comando
 - comandos.c, 16
 - comandos.h, 18
- executaCriaJogo
 - comandos.c, 16
 - comandos.h, 18
- executaGravaJogo
 - comandos.c, 17
 - comandos.h, 18
- executaJogada
 - comandos.c, 17
 - comandos.h, 19
- ficheiroValido
 - tabuleiro.c, 25
 - tabuleiro.h, 28
- freeTab
 - tabuleiro.c, 25
 - tabuleiro.h, 28
- gravaJogo
 - tabuleiro.c, 25
 - tabuleiro.h, 29
- jogada
 - tabuleiro.c, 26
 - tabuleiro.h, 29
- letra
 - elemento, 5
- main
 - main.c, 23
- main.c, 23
 - main, 23
- mensagem
 - erro, 6
- mensagem_de_erro
 - erro.c, 20
 - erro.h, 21
- pds
 - ajuda.c, 9
 - ajuda.h, 13
- pis
 - ajuda.c, 9
 - ajuda.h, 13
- printTabuleiro
 - tabuleiro.c, 26
 - tabuleiro.h, 29
- snd
 - ajuda.c, 9
 - ajuda.h, 13
- tabuleiro.c, 24
 - criarTabuleiro, 24
 - dimensao, 25
 - ficheiroValido, 25
 - freeTab, 25
 - gravaJogo, 25
 - jogada, 26
 - printTabuleiro, 26
- tabuleiro.h, 27
 - criarTabuleiro, 27
 - dimensao, 28
 - Elem, 27
 - Elemento, 27
 - ficheiroValido, 28
 - freeTab, 28
 - gravaJogo, 29
 - jogada, 29
 - printTabuleiro, 29
- trp
 - ajuda.c, 10
 - ajuda.h, 14
- vb
 - ajuda.c, 10
 - ajuda.h, 14
- vl
 - ajuda.c, 10
 - ajuda.h, 14
- vp
 - ajuda.c, 10
 - ajuda.h, 15