

Projeto II - relatório
v1.0

Gerado por Doxygen 1.8.20

1 Trabalho 2 de Estrutura de Dados	1
2 Índice dos namespaces	3
2.1 Lista de namespaces	3
3 Índice dos componentes	5
3.1 Lista de componentes	5
4 Índice dos ficheiros	7
4.1 Lista de ficheiros	7
5 Documentação dos namespaces	9
5.1 Referência ao namespace structures	9
5.1.1 Descrição detalhada	9
5.1.2 Documentação das funções	9
5.1.2.1 <code>init()</code>	9
5.1.2.2 <code>insert()</code>	9
6 Documentação da classe	11
6.1 Referência à estrutura <code>structures::NoTrie</code>	11
6.1.1 Descrição detalhada	11
6.1.2 Documentação dos dados membro	11
6.1.2.1 filhos	11
6.1.2.2 <code>pos</code>	11
6.1.2.3 <code>size</code>	12
7 Documentação do ficheiro	13
7.1 Referência ao ficheiro <code>dic.h</code>	13
7.1.1 Documentação das funções	13
7.1.1.1 <code>parseDict()</code>	13
7.1.2 Documentação das variáveis	14
7.1.2.1 <code>nextPos</code>	14
7.2 Referência ao ficheiro <code>main.cpp</code>	14
7.2.1 Documentação das funções	14
7.2.1.1 <code>main()</code>	15
7.2.1.2 <code>outputResults()</code>	15
7.3 Referência ao ficheiro <code>README.md</code>	15
7.4 Referência ao ficheiro <code>trie.h</code>	15
7.5 Referência ao ficheiro <code>trie.inc</code>	16
7.5.1 Documentação das funções	16
7.5.1.1 <code>countSufixes()</code>	16
7.5.1.2 <code>find()</code>	17
7.5.1.3 <code>init()</code>	17
7.5.1.4 <code>insert()</code>	17

Capítulo 1

Trabalho 2 de Estrutura de Dados

Um programa que lê um arquivo de texto com um dicionário e armazena os termos de um jeito bem inteligente usando uma Trie.

Autores

André Fernandes Hans Buss

Capítulo 2

Índice dos namespaces

2.1 Lista de namespaces

Lista dos namespaces com uma breve descrição:

structures	
Nó de uma Trie	9

Capítulo 3

Índice dos componentes

3.1 Lista de componentes

Lista de classes, estruturas, uniões e interfaces com uma breve descrição:

structures::NoTrie	11
--	----

Capítulo 4

Índice dos ficheiros

4.1 Lista de ficheiros

Lista de todos os ficheiros com uma breve descrição:

dic.h	13
main.cpp	14
trie.h	15
trie.inc	16

Capítulo 5

Documentação dos namespaces

5.1 Referência ao namespace structures

Nó de uma Trie.

Componentes

- struct `NoTrie`

Funções

- struct `NoTrie` * `initi` ()
- void `insert` (struct `NoTrie` *`root`, std::string `key`, unsigned long `pos`, unsigned long `size`)

5.1.1 Descrição detalhada

Nó de uma Trie.

5.1.2 Documentação das funções

5.1.2.1 `initi()`

```
struct NoTrie* structures::initi ( )
```

5.1.2.2 `insert()`

```
void structures::insert (
    struct NoTrie * root,
    std::string key,
    unsigned long pos,
    unsigned long size )
```


Capítulo 6

Documentação da classe

6.1 Referência à estrutura `structures::NoTrie`

```
#include <trie.h>
```

Atributos Públicos

- struct `NoTrie` * `filhos` [26]
- unsigned long `size`
- unsigned long `pos`

6.1.1 Descrição detalhada

Definido na linha 9 do ficheiro `trie.h`.

6.1.2 Documentação dos dados membro

6.1.2.1 `filhos`

```
struct NoTrie* structures::NoTrie::filhos[26]
```

Definido na linha 10 do ficheiro `trie.h`.

6.1.2.2 `pos`

```
unsigned long structures::NoTrie::pos
```

Definido na linha 11 do ficheiro `trie.h`.

Referenciado por `outputResults()`.

6.1.2.3 size

```
unsigned long structures::NoTrie::size
```

Definido na linha 11 do ficheiro trie.h.

Referenciado por `outputResults()`.

A documentação para esta estrutura foi gerada a partir do seguinte ficheiro:

- [trie.h](#)

Capítulo 7

Documentação do ficheiro

7.1 Referência ao ficheiro dic.h

```
#include "trie.h"
```

Funções

- `NoTrie * parseDict (std::string filename)`
Gera uma Trie a partir de um arquivo de dicionário.

Variáveis

- `int nextPos = 0`

7.1.1 Documentação das funções

7.1.1.1 parseDict()

```
NoTrie* parseDict (  
    std::string filename )
```

Gera uma Trie a partir de um arquivo de dicionário.

Parâmetros

<i>string</i>	Caminho para o arquivo
---------------	------------------------

Retorna

Nó raiz da trie gerada pelo arquivo

Definido na linha 16 do ficheiro dic.h.

Referenciado por main().

7.1.2 Documentação das variáveis

7.1.2.1 nextPos

```
int nextPos = 0
```

Definido na linha 6 do ficheiro dic.h.

Referenciado por parseDict().

7.2 Referência ao ficheiro main.cpp

```
#include <fstream>
#include <iostream>
#include <string>
#include "trie.h"
#include "dic.h"
```

Funções

- void [outputResults](#) ([structures::NoTrie](#) *node, std::string word)
Printa a resposta correspondente ao nó e à palavra buscada.
- int [main](#) ()
Executa o programa, e lê as entradas.

7.2.1 Documentação das funções

7.2.1.1 main()

```
int main ( )
```

Executa o programa, e lê as entradas.

Retorna

- 0 se o programa funcionar;
- 1 se não foi possível abrir o arquivo;

Definido na linha 37 do ficheiro main.cpp.

7.2.1.2 outputResults()

```
void outputResults (
    structures::NoTrie * node,
    std::string word )
```

Printa a resposta correspondente ao nó e à palavra buscada.

Parâmetros

<i>NoTrie</i>	nó correspondente a palavra buscada
<i>string</i>	palavra buscada

Definido na linha 15 do ficheiro main.cpp.

Referenciado por main().

7.3 Referência ao ficheiro README.md

7.4 Referência ao ficheiro trie.h

```
#include "trie.inc"
```

Componentes

- struct `structures::NoTrie`

Namespaces

- `structures`
Nó de uma Trie.

Funções

- struct NoTrie * [structures::init](#) ()
- void [structures::insert](#) (struct NoTrie *root, std::string key, unsigned long pos, unsigned long size)

7.5 Referência ao ficheiro trie.inc

```
#include <iostream>
```

Funções

- struct NoTrie * [init](#) ()
Cria um NoTrie.
- void [insert](#) (struct NoTrie *root, std::string key, unsigned long pos, unsigned long size)
Insere valores em uma Trie.
- NoTrie * [find](#) (struct NoTrie *root, std::string key)
encontra o último nó de uma palavra de uma Trie.
- int [countSufixes](#) (struct NoTrie *root)
Conta os sufixos de um nó.

7.5.1 Documentação das funções

7.5.1.1 countSufixes()

```
int countSufixes (  
    struct NoTrie * root )
```

Conta os sufixos de um nó.

Parâmetros

<i>NoTrie</i>	raiz da trie.
---------------	---------------

Retorna

Quantidade de sufixos de uma palavra no dicionário.

Definido na linha 78 do ficheiro trie.inc.

Referenciado por `outputResults()`.

7.5.1.2 find()

```
NoTrie* find (
    struct NoTrie * root,
    std::string key )
```

encontra o último nó de uma palavra de uma Trie.

Parâmetros

<i>NoTrie</i>	raiz da trie
<i>string</i>	palavra

Retorna

nó desejado

Definido na linha 55 do ficheiro trie.inc.

Referenciado por main().

7.5.1.3 init()

```
struct NoTrie* init ( )
```

Cria um NoTrie.

Retorna

Nó inicializado.

Definido na linha 8 do ficheiro trie.inc.

Referenciado por find(), insert() e parseDict().

7.5.1.4 insert()

```
void insert (
    struct NoTrie * root,
    std::string key,
    unsigned long pos,
    unsigned long size )
```

Insere valores em uma Trie.

Parâmetros

<i>NoTrie</i>	raiz da trie
<i>string</i>	palavra
<i>long</i>	posição no dicionário
<i>long</i>	tamanho da linha

Definido na linha 30 do ficheiro trie.inc.

Referenciado por parseDict().

Índice

- countSufixes
 - trie.inc, [16](#)
- dic.h, [13](#)
 - nextPos, [14](#)
 - parseDict, [13](#)
- filhos
 - structures::NoTrie, [11](#)
- find
 - trie.inc, [16](#)
- init
 - trie.inc, [17](#)
- initi
 - structures, [9](#)
- insert
 - structures, [9](#)
 - trie.inc, [17](#)
- main
 - main.cpp, [14](#)
- main.cpp, [14](#)
 - main, [14](#)
 - outputResults, [15](#)
- nextPos
 - dic.h, [14](#)
- outputResults
 - main.cpp, [15](#)
- parseDict
 - dic.h, [13](#)
- pos
 - structures::NoTrie, [11](#)
- README.md, [15](#)
- size
 - structures::NoTrie, [11](#)
- structures, [9](#)
 - initi, [9](#)
 - insert, [9](#)
- structures::NoTrie, [11](#)
 - filhos, [11](#)
 - pos, [11](#)
 - size, [11](#)
- trie.h, [15](#)
- trie.inc, [16](#)
- countSufixes, [16](#)
- find, [16](#)
- init, [17](#)
- insert, [17](#)