Huffman

Generated by Doxygen 1.9.3

Chapter 1

Class Index

1.1 Class List

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

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

Class Documentation

3.1 Node Struct Reference

`#include <node.h>`

Public Member Functions

- Node (int frequency)
- Node (int frequency, char character)

Public Attributes

- int frequency
- char character
- std::string code
- Node * left = NULL
- Node * right = NULL

3.1.1 Detailed Description

Struktura ta to wezel przechowujacy informacje o czestotliwosci okreslonego znaku oraz jego kodu, stanowi element drzewa Huffmana.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 Node() [1/2]

```
Node::Node (
            int frequency )
```

Domyslny konstruktor wezla pustego.

Parameters

frequency	czestotliwosc wystepowania znakow

3.1.2.2 Node() [2/2]

```
Node::Node (
            int frequency,
            char character )
```

Domyslny konstruktor wezla ze znakiem.

Parameters

frequency	czestotliwosc wystepowania znaku
character	znak dla jakiego jest przypisana czestotliwosc

3.1.3 Member Data Documentation

3.1.3.1 character

```
char Node::character
```

3.1.3.2 code

```
std::string Node::code
```

3.1.3.3 frequency

```
int Node::frequency
```

3.1.3.4 left

```
Node* Node::left = NULL
```

3.1.3.5 right

```
Node* Node::right = NULL
```

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

- node.h
- node.cpp

3.2 NodeComparator Struct Reference

```
#include <node.h>
```

Public Member Functions

- bool operator() (const Node *leftNode, const Node *rightNode)

3.2.1 Detailed Description

Struktura ta sluzy jedynie do porownywania ze soba dwoch wezlow struktury 'Node', jej implementacja jest wymagana przez std::priority_queue.

3.2.2 Member Function Documentation

3.2.2.1 operator()()

```
bool NodeComparator::operator() (
            const Node * leftNode,
            const Node * rightNode )
```

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

- node.h
- node.cpp

Chapter 4

File Documentation

4.1 functions.cpp File Reference

```
#include <iostream>
#include <fstream>
#include "functions.h"
#include "utils.h"
```

Functions

- void makeFrequencyDictionary (const std::string &text, std::map< char, int > &frequency)
- void loadFrequencyDictionary (const std::string &dict_filepath, std::map< char, int > &frequency)
- void saveFrequencyDictionary (const std::string &dict_filepath, const std::map< char, int > &frequency)
- void makeHeap (std::priority_queue< Node *, std::vector< Node * >, NodeComparator > &minBinaryHeap, const std::map< char, int > &frequency)
- void calculateCodes (Node *root, std::map< char, std::string > &dictionary, std::string code)
- void encode (const std::string &input_filepath, const std::string &output_filepath, std::map< char, std::string > &dictionary)
- void compress (const std::string &input_filepath, const std::string &output_filepath, const std::string &dict_↩ filepath)
- void decompress (const std::string &input_filepath, const std::string &output_filepath, const std::string &dict↩ _filepath)
- void decode (const std::string &input_filepath, const std::string &output_filepath, std::map< char, std::string > &dictionary, std::map< char, int > &frequency)

4.1.1 Function Documentation

4.1.1.1 calculateCodes()

```
void calculateCodes (
            Node * root,
            std::map< char, std::string > & dictionary,
            std::string code = "" )
```

Oblicza kody dla kazdego unikatowego znaku przemieszczajac sie po drzewie Huffmana okreslonym w zmiennej 'minBinaryHeap'.

Parameters

root	wskaznik na koniec kolejki priorytetowej
dictionary	referencja do slownika
code	argument opcjonalny do wewnetrznego zastosowania

4.1.1.2 compress()

```
void compress (
            const std::string & input_filepath,
            const std::string & output_filepath,
            const std::string & dict_filepath )
```

Interfejs sluzacy do kompresji pliku wejsciowego do wyjsciowego i zapisania pliku slownikowego za pomoca algorytmu Huffmana.

Parameters

intput_filepath	sciezka do pliku wejsciowego
output_filepath	sciezka do pliku wyjsciowego
dict_filepath	sciezka do pliku slownikowego

4.1.1.3 decode()

```
void decode (
            const std::string & input_filepath,
            const std::string & output_filepath,
            std::map< char, std::string > & dictionary,
            std::map< char, int > & frequency )
```

Odkodowuje plik ze sciezki okreslonej w zmiennej klasowej 'input_filepath' do sciezki wyjsciowej 'output_filepath' funkcja ta jedynie opakowuje funkcje prywatne do prostej uzywalnej formy.

Parameters

intput_filepath	sciezka do pliku wejsciowego
output_filepath	sciezka do pliku wyjsciowego
dictionary	referencja do slownika
frequency	referencja do frekwencji wystepowania znakow

4.1.1.4 decompress()

```
void decompress (
```

```
            const std::string & input_filepath,
            const std::string & output_filepath,
            const std::string & dict_filepath )
```

Interfejs sluzacy do dekompresji pliku wejsciowego do wyjsciowego przy uzyciu okreslonego slownika.

Parameters

intput_filepath	sciezka do pliku wejsciowego
output_filepath	sciezka do pliku wyjsciowego
dict_filepath	sciezka do pliku slownikowego

4.1.1.5 encode()

```
void encode (
            const std::string & input_filepath,
            const std::string & output_filepath,
            std::map< char, std::string > & dictionary )
```

Koduje plik ze sciezki okreslonej w zmiennej klasowej 'input_filepath' do sciezki wyjsciowej 'output_filepath', funkcja ta jedynie opakowuje funkcje do prostej uzywalnej formy.

Parameters

intput_filepath	sciezka do pliku wejsciowego
output_filepath	sciezka do pliku wyjsciowego
dictionary	referencja do slownika

4.1.1.6 loadFrequencyDictionary()

```
void loadFrequencyDictionary (
            const std::string & dict_filepath,
            std::map< char, int > & frequency )
```

Laduje slownik unikatowych znakow z pliku, sciezka do pliku jest okreslona w zmiennej 'dict_filepath' klasy.

Parameters

dict_filepath	sciezka do pliku slownikowego
frequency	referencja do frekwencji wystepowania znakow

4.1.1.7 makeFrequencyDictionary()

```
void makeFrequencyDictionary (
            const std::string & text,
            std::map< char, int > & frequency )
```

Tworzy slownik unikatowych znakow na podstawie przeslanego tekstu oraz zlicza ich wystapienie.

Parameters

text	referencja do tekstu do wyszukania unikatowych znakow
frequency	referencja do frekwencji wystepowania znakow

4.1.1.8 makeHeap()

```
void makeHeap (
            std::priority_queue< Node *, std::vector< Node * >, NodeComparator > & minBinaryHeap,
            const std::map< char, int > & frequency )
```

Tworzy drzewo Huffmana w postaci kolejki priorytetowej <priority_queue> oraz zapelnia je danymi ze slownika.

Parameters

minBinaryHeap	referencja do kolejki priorytetowej
frequency	referencja do frekwencji wystepowania znakow

4.1.1.9 saveFrequencyDictionary()

```
void saveFrequencyDictionary (
            const std::string & dict_filepath,
            const std::map< char, int > & frequency )
```

Zapisuje slownik unikatowych znakow do pliku, sciezka do pliku jest okreslona w zmiennej 'dict_filepath' klasy.

Parameters

dict_filepath	sciezka do pliku slownikowego
frequency	referencja do frekwencji wystepowania znakow

4.2 functions.h File Reference

```
#include <string>
```

```
#include <map>
#include <queue>
#include <vector>
#include "node.h"
```

Functions

- void makeFrequencyDictionary (const std::string &text, std::map< char, int > &frequency)
- void loadFrequencyDictionary (const std::string &dict_filepath, std::map< char, int > &frequency)
- void saveFrequencyDictionary (const std::string &dict_filepath, const std::map< char, int > &frequency)
- void makeHeap (std::priority_queue< Node *, std::vector< Node * >, NodeComparator > &minBinaryHeap, const std::map< char, int > &frequency)
- void calculateCodes (Node *root, std::map< char, std::string > &dictionary, std::string code="")
- void encode (const std::string &input_filepath, const std::string &output_filepath, std::map< char, std::string > &dictionary)
- void decode (const std::string &input_filepath, const std::string &output_filepath, std::map< char, std::string > &dictionary, std::map< char, int > &frequency)
- void compress (const std::string &input_filepath, const std::string &output_filepath, const std::string &dict_↩ filepath)
- void decompress (const std::string &input_filepath, const std::string &output_filepath, const std::string &dict↩ _filepath)

4.2.1 Function Documentation

4.2.1.1 calculateCodes()

```
void calculateCodes (
            Node * root,
            std::map< char, std::string > & dictionary,
            std::string code = "" )
```

Oblicza kody dla kazdego unikatowego znaku przemieszczajac sie po drzewie Huffmana okreslonym w zmiennej 'minBinaryHeap'.

Parameters

root	wskaznik na koniec kolejki priorytetowej
dictionary	referencja do slownika
code	argument opcjonalny do wewnetrznego zastosowania

4.2.1.2 compress()

```
void compress (
            const std::string & input_filepath,
```

```
            const std::string & output_filepath,
            const std::string & dict_filepath )
```

Interfejs sluzacy do kompresji pliku wejsciowego do wyjsciowego i zapisania pliku slownikowego za pomoca algorytmu Huffmana.

Parameters

intput_filepath	sciezka do pliku wejsciowego
output_filepath	sciezka do pliku wyjsciowego
dict_filepath	sciezka do pliku slownikowego

4.2.1.3 decode()

```
void decode (
            const std::string & input_filepath,
            const std::string & output_filepath,
            std::map< char, std::string > & dictionary,
            std::map< char, int > & frequency )
```

Odkodowuje plik ze sciezki okreslonej w zmiennej klasowej 'input_filepath' do sciezki wyjsciowej 'output_filepath' funkcja ta jedynie opakowuje funkcje prywatne do prostej uzywalnej formy.

Parameters

intput_filepath	sciezka do pliku wejsciowego
output_filepath	sciezka do pliku wyjsciowego
dictionary	referencja do slownika
frequency	referencja do frekwencji wystepowania znakow

4.2.1.4 decompress()

```
void decompress (
            const std::string & input_filepath,
            const std::string & output_filepath,
            const std::string & dict_filepath )
```

Interfejs sluzacy do dekompresji pliku wejsciowego do wyjsciowego przy uzyciu okreslonego slownika.

Parameters

intput_filepath	sciezka do pliku wejsciowego
output_filepath	sciezka do pliku wyjsciowego
dict_filepath	sciezka do pliku slownikowego

4.2.1.5 encode()

```
void encode (
            const std::string & input_filepath,
            const std::string & output_filepath,
            std::map< char, std::string > & dictionary )
```

Koduje plik ze sciezki okreslonej w zmiennej klasowej 'input_filepath' do sciezki wyjsciowej 'output_filepath', funkcja ta jedynie opakowuje funkcje do prostej uzywalnej formy.

Parameters

intput_filepath	sciezka do pliku wejsciowego
output_filepath	sciezka do pliku wyjsciowego
dictionary	referencja do slownika

4.2.1.6 loadFrequencyDictionary()

```
void loadFrequencyDictionary (
            const std::string & dict_filepath,
            std::map< char, int > & frequency )
```

Laduje slownik unikatowych znakow z pliku, sciezka do pliku jest okreslona w zmiennej 'dict_filepath' klasy.

Parameters

dict_filepath	sciezka do pliku slownikowego
frequency	referencja do frekwencji wystepowania znakow

4.2.1.7 makeFrequencyDictionary()

```
void makeFrequencyDictionary (
            const std::string & text,
            std::map< char, int > & frequency )
```

Tworzy slownik unikatowych znakow na podstawie przeslanego tekstu oraz zlicza ich wystapienie.

Parameters

text	referencja do tekstu do wyszukania unikatowych znakow
frequency	referencja do frekwencji wystepowania znakow

4.2.1.8 makeHeap()

```
void makeHeap (
            std::priority_queue< Node *, std::vector< Node * >, NodeComparator > & minBinaryHeap,
            const std::map< char, int > & frequency )
```

Tworzy drzewo Huffmana w postaci kolejki priorytetowej <priority_queue> oraz zapelnia je danymi ze slownika.

Parameters

minBinaryHeap	referencja do kolejki priorytetowej
frequency	referencja do frekwencji wystepowania znakow

4.2.1.9 saveFrequencyDictionary()

```
void saveFrequencyDictionary (
            const std::string & dict_filepath,
            const std::map< char, int > & frequency )
```

Zapisuje slownik unikatowych znakow do pliku, sciezka do pliku jest okreslona w zmiennej 'dict_filepath' klasy.

Parameters

dict_filepath	sciezka do pliku slownikowego
frequency	referencja do frekwencji wystepowania znakow

4.3 functions.h

Go to the documentation of this file.

```
1 #ifndef FUNCTIONS_H
2 #define FUNCTIONS_H
3
4 #include <string>
5 #include <map>
6 #include <queue>
7 #include <vector>
8
9 #include "node.h"
10
18 void makeFrequencyDictionary(const std::string &text, std::map<char, int> &frequency);
19
28 void loadFrequencyDictionary(const std::string &dict_filepath, std::map<char, int> &frequency);
29
38 void saveFrequencyDictionary(const std::string &dict_filepath, const std::map<char, int> &frequency);
39
47 void makeHeap(std::priority_queue<Node *, std::vector<Node *>, NodeComparator> &minBinaryHeap, const
       std::map<char, int> &frequency);
48
57 void calculateCodes(Node *root, std::map<char, std::string> &dictionary, std::string code = "");
58
```

```
69 void encode(const std::string &input_filepath, const std::string &output_filepath, std::map<char,
       std::string> &dictionary);
70
82 void decode(const std::string &input_filepath, const std::string &output_filepath, std::map<char,
       std::string> &dictionary, std::map<char, int> &frequency);
83
93 void compress(const std::string &input_filepath, const std::string &output_filepath, const std::string
       &dict_filepath);
94
104 void decompress(const std::string &input_filepath, const std::string &output_filepath, const std::string
       &dict_filepath);
105
106 #endif
```

4.4 main.cpp File Reference

```
#include <iostream>
#include <unistd.h>
#include "utils.h"
#include "functions.h"
```

Functions

- void printHelp ()
- int main (int argc, char ∗argv[ ])

4.4.1 Function Documentation

4.4.1.1 main()

```
int main (
            int argc,
            char * argv[] )
```

4.4.1.2 printHelp()

```
void printHelp ( )
```

4.5 node.cpp File Reference

```
#include "node.h"
```

4.6 node.h File Reference

```
#include <iostream>
```

Classes

- struct Node
- struct NodeComparator

4.7 node.h

Go to the documentation of this file.

```
1 #ifndef NODE_H
2 #define NODE_H
3
4 #include <iostream>
5
11 struct Node
12 {
13 public:
14     int frequency;
15     char character;
16     std::string code;
17
18     Node *left = NULL;
19     Node *right = NULL;
20
26     Node(int frequency);
27
34     Node(int frequency, char character);
35 };
36
42 struct NodeComparator
43 {
44 public:
45     bool operator()(const Node *leftNode, const Node *rightNode);
46 };
47
48 #endif
```

4.8 utils.cpp File Reference

```
#include "utils.h"
```

Functions

- std::string getFileContent (const std::string filepath)

4.8.1 Function Documentation

4.8.1.1 getFileContent()

```
std::string getFileContent (
            const std::string filepath )
```

Funkcja wczytuje zawartosc pliku okreslonego w sciezce.

Parameters

filepath	sciezka do pliku

Returns

funkcja zwraca std::string zawartosci

4.9 utils.h File Reference

```
#include <iostream>
#include <fstream>
```

Functions

- std::string getFileContent (const std::string filepath)

4.9.1 Function Documentation

4.9.1.1 getFileContent()

```
std::string getFileContent (
            const std::string filepath )
```

Funkcja wczytuje zawartosc pliku okreslonego w sciezce.

Parameters

filepath	sciezka do pliku

Returns

funkcja zwraca std::string zawartosci

4.10 utils.h

Go to the documentation of this file.

```
1 #ifndef UTILS_H
2 #define UTILS_H
3
4 #include <iostream>
5 #include <fstream>
6
13 std::string getFileContent(const std::string filepath);
14
15 #endif
```

Index