DCT

Generated by Doxygen 1.9.5

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 documented files with brief descriptions:

Chapter 3

Class Documentation

3.1 gate Struct Reference

Public Attributes

- int **inx**
- int **iny**
- gatename **name**

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

- DCT/gates.h

Chapter 4

File Documentation

4.1 DCT/gates.cpp File Reference

```
#include <iostream>
#include <fstream>
#include <unordered_map>
#include <sstream>
#include <string>
#include "gates.h"
```

Functions

- void load_data (string &gatesname, unordered_map< int, gate > &gates, vector< int > &outputnodes, int &inputamount)
- void startevaluation (string &line, unordered_map< int, gate > &gates, ofstream &output, vector< int > &outputnodes, int &inputamount)
- bool evaluate_gate (const int &gatenum, unordered_map< int, gate > &gates, ofstream &output)

4.1.1 Function Documentation

4.1.1.1 evaluate_gate()

```
bool evaluate_gate (
            const int & gatenum,
            unordered_map< int, gate > & gates,
            ofstream & output )
```

The function evaluates logic gates in order to find the value of a given node.

Parameters

gatenum	Number of the node that is being evaluated.
gates	Unordered map which holds gates and inputs.
output	Output file.

Returns

The function returns the value of the node.

Author

Jakub Knapik

4.1.1.2 load_data()

```
void load_data (
            string & gatesname,
            unordered_map< int, gate > & gates,
            vector< int > & outputnodes,
            int & inputamount )
```

The function loads data from files.

Parameters

gatesname	Name of the input file with input gates.
gates	Unordered map which holds gates and inputs.
outputnodes	Vector which holds output nodes.
inputamount	Counter holding the number of input nodes.

Author

Jakub Knapik

4.1.1.3 startevaluation()

```
void startevaluation (
            string & line,
            unordered_map< int, gate > & gates,
            ofstream & output,
            vector< int > & outputnodes,
            int & inputamount )
```

The function starts the evaluation of the output nodes, and outputs the result.

Parameters

line	Values for input nodes.
gates	Unordered map which holds gates and inputs.
output	Output file.
outputnodes	Vector which holds output nodes.
inputamount	Counter holding the number of input nodes.

Author

Jakub Knapik

4.2 DCT/gates.h File Reference

```
#include <iostream>
#include <unordered_map>
```

Classes

- struct gate

Enumerations

- enum **gatename** {
  IN , **AND** , **NAND** , **OR** ,
  NOR , **XOR** , **XNOR** , **NEG** }

Functions

- void load_data (string &gatesname, unordered_map< int, gate > &gates, vector< int > &outputnodes, int &inputamount)
- void startevaluation (string &line, unordered_map< int, gate > &gates, ofstream &output, vector< int > &outputnodes, int &inputamount)
- bool evaluate_gate (const int &gatenum, unordered_map< int, gate > &gates, ofstream &output)

4.2.1 Function Documentation

4.2.1.1 evaluate_gate()

```
bool evaluate_gate (
            const int & gatenum,
            unordered_map< int, gate > & gates,
            ofstream & output )
```

The function evaluates logic gates in order to find the value of a given node.

Parameters

gatenum	Number of the node that is being evaluated.
gates	Unordered map which holds gates and inputs.
output	Output file.

Returns

The function returns the value of the node.

Author

Jakub Knapik

4.2.1.2 load_data()

```
void load_data (
            string & gatesname,
            unordered_map< int, gate > & gates,
            vector< int > & outputnodes,
            int & inputamount )
```

The function loads data from files.

Parameters

gatesname	Name of the input file with input gates.
gates	Unordered map which holds gates and inputs.
outputnodes	Vector which holds output nodes.
inputamount	Counter holding the number of input nodes.

Author

Jakub Knapik

4.2.1.3 startevaluation()

```
void startevaluation (
            string & line,
            unordered_map< int, gate > & gates,
            ofstream & output,
            vector< int > & outputnodes,
            int & inputamount )
```

The function starts the evaluation of the output nodes, and outputs the result.

Parameters

line	Values for input nodes.
gates	Unordered map which holds gates and inputs.
output	Output file.
outputnodes	Vector which holds output nodes.
inputamount	Counter holding the number of input nodes.

Author

Jakub Knapik

4.3 gates.h

Go to the documentation of this file.

```
1
2 #include <iostream>
3 #include <unordered_map>
4 using namespace std;
5
6 #ifndef gates_h
7 #define gates_h
8 enum gatename{IN,AND,NAND,OR,NOR,XOR,XNOR,NEG};
9 struct gate
10 {
11     int inx;
12     int iny;
13     gatename name;
14 };
22 void load_data(string& gatesname, unordered_map<int, gate>& gates, vector<int>& outputnodes, int&
      inputamount);
31 void startevaluation(string& line, unordered_map<int, gate>& gates,ofstream& output, vector<int>&
      outputnodes, int& inputamount);
39 bool evaluate_gate(const int& gatenum,unordered_map<int, gate>& gates, ofstream& output);
40
41 #endif // GATES_H
```

4.4 DCT/main.cpp File Reference

```
#include <iostream>
#include <fstream>
#include <unordered_map>
#include <vector>
#include <string>
#include "gates.h"
```

Functions

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

Index