# Drzewo czerwono-czarne

Generated by Doxygen 1.8.7

Sat Apr 30 2016 15:52:52

# Contents

# Chapter 1

# Hierarchical Index

## 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

# Chapter 2

# Class Index

## 2.1 Class List

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

# Chapter 3

# File Index

## 3.1 File List

Here is a list of all files with brief descriptions:

# Chapter 4

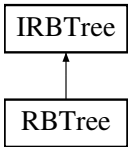
# Class Documentation

## 4.1 IRBTree Class Reference

Interfejs drzewa czerwono-czarnego.

`#include <itree.hh>`

Inheritance diagram for IRBTree:




### Public Member Functions

- virtual void **add** (const int &elem)=0
- virtual int **find** (const int &elem)=0
- virtual void **release** (**Node** ∗n)=0
- virtual void **rotate_left** (**Node** ∗n)=0
- virtual void **rotate_right** (**Node** ∗n)=0
- virtual ∼**IRBTree** ()

### 4.1.1 Detailed Description

Interfejs drzewa czerwono-czarnego.

Zawiera metody umożliwiające operacje na drzewie.

Definition at line 12 of file itree.hh.

### 4.1.2 Constructor & Destructor Documentation

#### 4.1.2.1 virtual IRBTree::∼IRBTree ( ) [inline], [virtual]

Definition at line 20 of file itree.hh.

### 4.1.3 Member Function Documentation

#### 4.1.3.1 virtual void IRBTree::add ( const int & *elem* ) [pure virtual]

Implemented in **RBTree** (p. 12).

#### 4.1.3.2 virtual int IRBTree::find ( const int & *elem* ) [pure virtual]

Implemented in **RBTree** (p. 12).

#### 4.1.3.3 virtual void IRBTree::release ( Node * *n* ) [pure virtual]

Implemented in **RBTree** (p. 12).

#### 4.1.3.4 virtual void IRBTree::rotate_left ( Node * *n* ) [pure virtual]

Implemented in **RBTree** (p. 13).

#### 4.1.3.5 virtual void IRBTree::rotate_right ( Node * *n* ) [pure virtual]

Implemented in **RBTree** (p. 13).

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

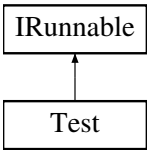
- **itree.hh**

## 4.2 IRunnable Class Reference

Interfejs klasy rozruchowej.

`#include <irunnable.hh>`

Inheritance diagram for IRunnable:




### Public Member Functions

- virtual void **run** (int Argc, char *Argv[])=0

### 4.2.1 Detailed Description

Interfejs klasy rozruchowej.

Zawiera metodę umożliwiającą uruchomienie programu.

Definition at line 9 of file irunnable.hh.

### 4.2.2 Member Function Documentation

#### 4.2.2.1 virtual void IRunnable::run ( int *Argc,* char * *Argv[]* ) [pure virtual]

Implemented in **Test** (p. 14).

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

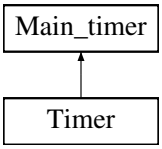
- **irunnable.hh**

## 4.3 Main_timer Class Reference

Interfejs stopera.

`#include <maintimer.hh>`

Inheritance diagram for Main_timer:




### Public Member Functions

- virtual long double **get_ms_time** ()=0
- virtual void **tim_start** ()=0
- virtual void **tim_stop** ()=0
- virtual long double **return_time** ()=0
- virtual ~**Main_timer** ()

### 4.3.1 Detailed Description

Interfejs stopera.

Zawiera metody umożliwiające mierzenie czasu.

Definition at line 9 of file maintimer.hh.

### 4.3.2 Constructor & Destructor Documentation

#### 4.3.2.1 virtual Main_timer::~Main_timer ( ) [inline], [virtual]

Definition at line 16 of file maintimer.hh.

### 4.3.3 Member Function Documentation

#### 4.3.3.1 virtual long double Main_timer::get_ms_time ( ) [pure virtual]

Implemented in **Timer** (p. 15).

#### 4.3.3.2 virtual long double Main_timer::return_time ( ) [pure virtual]

Implemented in **Timer** (p. 15).

#### 4.3.3.3 virtual void Main_timer::tim_start ( ) [pure virtual]

Implemented in **Timer** (p. 15).

#### 4.3.3.4 virtual void Main_timer::tim_stop ( ) [pure virtual]

Implemented in **Timer** (p. 15).

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

- **maintimer.hh**

## 4.4 Node Class Reference

Klasa węzła.

`#include <tree.hh>`

### Private Attributes

- int **elem**
- char **color**
- **Node** * **father**
- **Node** * **left_son**
- **Node** * **right_son**

### Friends

- class **RBTree**

### 4.4.1 Detailed Description

Klasa węzła.

Zawiera wskaźniki na ojca, lewego syna i prawego syna.

Definition at line 11 of file tree.hh.

### 4.4.2 Friends And Related Function Documentation

#### 4.4.2.1 friend class RBTree [friend]

Definition at line 12 of file tree.hh.

### 4.4.3 Member Data Documentation

#### 4.4.3.1 char Node::color [private]

Definition at line 15 of file tree.hh.

#### 4.4.3.2 int Node::elem [private]

Definition at line 14 of file tree.hh.

#### 4.4.3.3 Node* Node::father [private]

Definition at line 16 of file tree.hh.

#### 4.4.3.4 Node* Node::left_son [private]

Definition at line 17 of file tree.hh.

#### 4.4.3.5 Node* Node::right_son [private]

Definition at line 18 of file tree.hh.

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

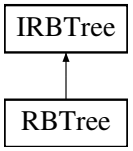
- **tree.hh**

## 4.5 RBTree Class Reference

Klasa drzewa czerwono czarnego.

`#include <tree.hh>`

Inheritance diagram for RBTree:




### Public Member Functions

- **RBTree** ()
- ~**RBTree** ()
- void **add** (const int &elem)
- int **find** (const int &elem)
- void **release** (**Node** *n)
- void **rotate_left** (**Node** *n)
- void **rotate_right** (**Node** *n)

### Private Attributes

- **Node** * **root** =NULL
- **Node** * **sentinel** =NULL

### 4.5.1 Detailed Description

Klasa drzewa czerwono czarnego.

Zawiera metody umożliwiające operacje na tablicy.

Definition at line 27 of file tree.hh.

### 4.5.2 Constructor & Destructor Documentation

#### 4.5.2.1 RBTree::RBTree ( )

Definition at line 3 of file tree.cpp.

#### 4.5.2.2 RBTree::∼RBTree ( )

Definition at line 14 of file tree.cpp.

### 4.5.3 Member Function Documentation

#### 4.5.3.1 void RBTree::add ( const int & *elem* ) [virtual]

Funkcja dodająca element do drzewa.

**Parameters**

| | | |
|---|---|---|
| in | *wartość* | typu int |

Implements **IRBTree** (p. 8).

Definition at line 88 of file tree.cpp.

#### 4.5.3.2 int RBTree::find ( const int & *elem* ) [virtual]

Funkcja wyszukująca element w drzewie.

**Parameters**

| | | |
|---|---|---|
| in | *wartość* | typu int |

**Returns**

znaleziona wartość typu int

Implements **IRBTree** (p. 8).

Definition at line 170 of file tree.cpp.

#### 4.5.3.3 void RBTree::release ( Node ∗ *n* ) [virtual]

Funkcja usuwająca drzewo lub poddrzewo.

**Parameters**

| | | |
|---|---|---|
| in | *węzeł* | startowy |

Implements **IRBTree** (p. 8).

Definition at line 20 of file tree.cpp.

#### 4.5.3.4 void RBTree::rotate_left ( Node * *n* ) [virtual]

Funkcja wykonująca rotację w lewo względem podanego węzła.

**Parameters**

| | | |
|---|---|---|
| in | *węzeł* | rotacyjny |

Implements **IRBTree** (p. 8).

Definition at line 30 of file tree.cpp.

#### 4.5.3.5 void RBTree::rotate_right ( Node * *n* ) [virtual]

Funkcja wykonująca rotację w prawo względem podanego węzła.

**Parameters**

| | | |
|---|---|---|
| in | *węzeł* | rotacyjny |

Implements **IRBTree** (p. 8).

Definition at line 59 of file tree.cpp.

### 4.5.4 Member Data Documentation

#### 4.5.4.1 Node* RBTree::root =NULL [private]

Definition at line 30 of file tree.hh.

#### 4.5.4.2 Node* RBTree::sentinel =NULL [private]

Definition at line 31 of file tree.hh.

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

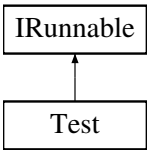
- **tree.hh**
- **tree.cpp**

## 4.6 Test Class Reference

Klasa rozruchowa.

`#include <test.hh>`

Inheritance diagram for Test:



### Public Member Functions

- void **run** (int Argc, char ∗Argv[])

### 4.6.1 Detailed Description

Klasa rozruchowa.

Zawiera metodę umożliwiającą uruchomienie programu.

Definition at line 10 of file test.hh.

### 4.6.2 Member Function Documentation

#### 4.6.2.1 void Test::run ( int *Argc,* char ∗ *Argv[]* ) [virtual]

Implements **IRunnable** (p. 9).

Definition at line 6 of file test.cpp.

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

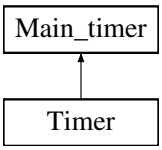
- **test.hh**
- **test.cpp**

## 4.7 Timer Class Reference

Klasa stopera.

`#include <timer.hh>`

Inheritance diagram for Timer:




### Public Member Functions

- long double **get_ms_time** ()
- void **tim_start** ()
- void **tim_stop** ()
- long double **return_time** ()
- ∼**Timer** ()

### Private Attributes

- long double **time_of_start**
- long double **time_of_stop**

### 4.7.1 Detailed Description

Klasa stopera.

Zawiera metody umożliwiające mierzenie czasu. Dokładny opis metod w dokumentacji projektu Lab2.

Definition at line 11 of file timer.hh.

### 4.7.2 Constructor & Destructor Documentation

#### 4.7.2.1 Timer::~Timer ( ) [inline]

Definition at line 21 of file timer.hh.

### 4.7.3 Member Function Documentation

#### 4.7.3.1 long double Timer::get_ms_time ( ) [virtual]

Implements **Main_timer** (p. 9).

Definition at line 25 of file timer.hh.

#### 4.7.3.2 long double Timer::return_time ( ) [virtual]

Implements **Main_timer** (p. 10).

Definition at line 47 of file timer.hh.

#### 4.7.3.3 void Timer::tim_start ( ) [virtual]

Implements **Main_timer** (p. 10).

Definition at line 35 of file timer.hh.

#### 4.7.3.4 void Timer::tim_stop ( ) [virtual]

Implements **Main_timer** (p. 10).

Definition at line 41 of file timer.hh.

### 4.7.4 Member Data Documentation

#### 4.7.4.1 long double Timer::time_of_start [private]

Definition at line 13 of file timer.hh.

#### 4.7.4.2 long double Timer::time_of_stop [private]

Definition at line 14 of file timer.hh.

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

- **timer.hh**

# Chapter 5

# File Documentation

## 5.1 irunnable.hh File Reference

### Classes

- class **IRunnable**

  *Interfejs klasy rozruchowej.*

## 5.2 itree.hh File Reference

```
#include "tree.hh"
```

### Classes

- class **IRBTree**

  *Interfejs drzewa czerwono-czarnego.*

## 5.3 main.cpp File Reference

```
#include "test.hh"
#include <ctime>
#include <cstdlib>
```

### Functions

- int **main** (int Argc, char ∗Argv[])

#### 5.3.1 Function Documentation

##### 5.3.1.1 int main ( int *Argc,* char ∗ *Argv[ ]* )

Definition at line 5 of file main.cpp.

## 5.4 maintimer.hh File Reference

### Classes

- class **Main_timer**

  *Interfejs stopera.*

## 5.5 test.cpp File Reference

```
#include "test.hh"
#include "tree.hh"
#include "timer.hh"
#include <iostream>
```

## 5.6 test.hh File Reference

```
#include "irunnable.hh"
```

### Classes

- class **Test**

  *Klasa rozruchowa.*

## 5.7 timer.hh File Reference

```
#include <sys/time.h>
#include "maintimer.hh"
```

### Classes

- class **Timer**

  *Klasa stopera.*

## 5.8 tree.cpp File Reference

```
#include "tree.hh"
```

## 5.9 tree.hh File Reference

```
#include "itree.hh"
#include <cstddef>
```

## Classes

- class **Node**

  *Klasa węzła.*
- class **RBTree**

  *Klasa drzewa czerwono czarnego.*

# Index