TreeDemo

0.1

Generated by Doxygen 1.8.7

Sun May 18 2014 22:50:56

Contents

Chapter 1

Class Index

1.1 Class List

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

Chapter 2

Class Documentation

2.1 BNode Class Reference

```
#include <BNode.h>
```

Public Member Functions

- BNode ()
- ∼BNode ()
- void printSubtree (ofstream &stream)

Public Attributes

- int values [size]
- BNode ∗ pointers [size+1]
- string vizcolors [size]

Static Public Attributes

- static const int size = 3
- static const int empty = -9999

2.1.1 Detailed Description

A node of a B-tree.

2.1.2 Constructor & Destructor Documentation

2.1.2.1 BNode::BNode ( )

Constructs new node with default (empty) values.

2.1.2.2 BNode::∼BNode ( )

Recursive post-order destructor.

2.1.3 Member Function Documentation

2.1.3.1 void BNode::printSubtree (ofstream & *stream*)

Prints subtree of the node into a Graphviz file.

Parameters

stream	Output stream.

2.1.4 Member Data Documentation

2.1.4.1 const int BNode::empty = -9999 `[static]`

Default value.

2.1.4.2 BNode∗ BNode::pointers[size+1]

Pointers to another nodes.

2.1.4.3 const int BNode::size = 3 `[static]`

Number of cells in the node.

2.1.4.4 int BNode::values[size]

Values of the node.

2.1.4.5 string BNode::vizcolors[size]

Colors of cells in the Graphviz visualization

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

- TreeDemo/BNode.h
- TreeDemo/BNode.cpp

2.2 BTree Class Reference

```
#include <BTree.h>
```

Public Member Functions

- BTree ()
- BTree (BNode ∗root)
- ∼BTree ()
- void visualize (string path, string dot)

2.2.1 Detailed Description

Structure of B-tree.

2.2.2 Constructor & Destructor Documentation

2.2.2.1 BTree::BTree ( )

Empty tree constructor.

2.2.2.2 BTree::BTree (BNode ∗ *root*) `[inline]`

Filled tree constructor.

Parameters

root	Node to set as the root.

2.2.2.3 BTree::∼BTree ( )

Default destructor.

2.2.3 Member Function Documentation

2.2.3.1 void BTree::visualize (string *path,* string *dot*)

Creates visualization of the current state of the tree using Graphviz.

Parameters

path	Where to save the PDF with visualization.
dot	Path to the Graphviz dot utility.

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

- TreeDemo/BTree.h
- TreeDemo/BTree.cpp

2.3 RBNode Class Reference

`#include <RBNode.h>`

Public Member Functions

- void printSubtree (ofstream &stream, RBNode ∗nil)
- RBNode ()
- RBNode (RBNode ∗nil, RBNode ∗parent, color_t color, int value)
- ∼RBNode ()

Public Attributes

- int **value**
- color_t color
- RBNode ∗ left
- RBNode ∗ right
- RBNode ∗ parent

Static Public Attributes

- static const color_t red = 0
- static const color_t black = 1
- static const int empty = -9999

2.3.1 Detailed Description

A node of a Red-black tree.

2.3.2 Constructor & Destructor Documentation

2.3.2.1 RBNode::RBNode ( ) [inline]

Constructs new node with default (empty) values.

2.3.2.2 RBNode::RBNode (RBNode $*$ *nil*, RBNode $*$ *parent*, color_t *color*, int *value*) [inline]

Constructs new node with preset values.

Parameters

nil	Nil node reference, will be set as left and right son.
parent	Reference to the parent node.
color	Color of the node.
value	Value of the node.

2.3.2.3 RBNode::∼RBNode ( )

Recursive post-order destructor, stops at the Nil node.

2.3.3 Member Function Documentation

2.3.3.1 void RBNode::printSubtree (ofstream & *stream*, RBNode $*$ *nil*)

Prints subtree of the node into a Graphviz file.

Parameters

stream	Output stream.
nil	Nil node to omit.

2.3.4 Member Data Documentation

2.3.4.1 const color_t RBNode::black = 1 [static]

Black color.

2.3.4.2 color_t RBNode::color

Color of the node.

2.3.4.3 const int RBNode::empty = -9999 `[static]`

Default value.

2.3.4.4 RBNode* RBNode::left

Left son of the node.

2.3.4.5 RBNode* RBNode::parent

Parent of the node.

2.3.4.6 const color_t RBNode::red = 0 `[static]`

Red color.

2.3.4.7 RBNode* RBNode::right

Right son of the node.

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

- TreeDemo/RBNode.h
- TreeDemo/RBNode.cpp

2.4 RBTree Class Reference

```
#include <RBTree.h>
```

Public Member Functions

- RBTree ()
- ~RBTree ()
- void add (int a)
- bool find (int a)
- void remove (int a)
- void visualize (string path, string dot)
- BTree btree ()

2.4.1 Detailed Description

Implementation of Red-black trees.

2.4.2 Constructor & Destructor Documentation

2.4.2.1 RBTree::RBTree ( )

Empty tree constructor.

2.4.2.2 RBTree::∼RBTree ( )

Default destructor.

2.4.3 Member Function Documentation

2.4.3.1 void RBTree::add (int *a*)

Adds new item into the tree. That is, if the tree doesn't already contain it.

Parameters

a	The item to add, positive integer expected.

2.4.3.2 BTree RBTree::btree ( )

Constructs a B-tree corresponding to the current state of the tree.

Returns

Analogical B-Tree.

2.4.3.3 bool RBTree::find (int *a*)

Searches for item in the tree.

Parameters

a	The item to search for, positive integer expected.

Returns

true if the tree contains a node with value equal to `a`, otherwise false

2.4.3.4 void RBTree::remove (int *a*)

Removes an item from the tree. If the tree doesn't contain this item, nothing happens.

Parameters

a	The item to remove, positive integer expected.

2.4.3.5 void RBTree::visualize (string *path*, string *dot*)

Creates visualization of the current state of the tree using Graphviz.

Parameters

path	Where to save the PDF with visualization.
dot	Path to the Graphviz dot utility.

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

- TreeDemo/RBTree.h
- TreeDemo/RBTree.cpp