

# AVL Binary Tree

version 2.1

Generated by Doxygen 1.8.9.1

Wed May 20 2015 18:44:41

# Chapter 1

## Class Index

### 1.1 Class List

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

<a href="#">AVLTree</a> . . . . .	<a href="#">5</a>
<a href="#">Node</a> . . . . .	<a href="#">7</a>



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

D:/STUDIA/PAMiSI/209226/AVL/ <a href="#">AVL.cpp</a> . . . . .	9
D:/STUDIA/PAMiSI/209226/AVL/ <a href="#">AVL.h</a> . . . . .	9
D:/STUDIA/PAMiSI/209226/AVL/ <a href="#">node.h</a> . . . . .	9
D:/STUDIA/PAMiSI/209226/AVL/test/ <a href="#">test_AVL.cpp</a> . . . . .	9



## Chapter 3

# Class Documentation

### 3.1 AVLTree Class Reference

```
#include <AVL.h>
```

#### Public Member Functions

- [AVLTree](#) ()
- virtual [~AVLTree](#) ()
- virtual [Node](#) \* [getRoot](#) ()
- virtual void [push](#) (const int &n)
- virtual void [printPreOrder](#) () const
- virtual void [preOrder](#) ([Node](#) \*pre) const
- virtual void [clear](#) ([Node](#) \*&tree)
- virtual void [singleRightRotate](#) ([Node](#) \*&n)
- virtual void [doubleRightRotate](#) ([Node](#) \*&n)
- virtual void [singleLeftRotate](#) ([Node](#) \*&n)
- virtual void [doubleLeftRotate](#) ([Node](#) \*&n)
- virtual bool [search](#) (const int &s)
- virtual int [avlHeight](#) ([Node](#) \*h)
- virtual int [max](#) (int v1, int v2)
- virtual void [print](#) ([Node](#) \*node, int level)

#### 3.1.1 Detailed Description

Definition at line 16 of file AVL.h.

#### 3.1.2 Constructor & Destructor Documentation

##### 3.1.2.1 AVLTree::AVLTree ( )

Definition at line 10 of file AVL.cpp.

##### 3.1.2.2 AVLTree::~~AVLTree ( ) [virtual]

Definition at line 16 of file AVL.cpp.

### 3.1.3 Member Function Documentation

**3.1.3.1** `int AVLTree::avlHeight ( Node * h )` [virtual]

Definition at line 67 of file AVL.cpp.

**3.1.3.2** `void AVLTree::clear ( Node *& tree )` [virtual]

Definition at line 112 of file AVL.cpp.

**3.1.3.3** `void AVLTree::doubleLeftRotate ( Node *& n )` [virtual]

Definition at line 56 of file AVL.cpp.

**3.1.3.4** `void AVLTree::doubleRightRotate ( Node *& n )` [virtual]

Definition at line 50 of file AVL.cpp.

**3.1.3.5** `Node * AVLTree::getRoot ( )` [virtual]

Definition at line 193 of file AVL.cpp.

**3.1.3.6** `int AVLTree::max ( int v1, int v2 )` [virtual]

Definition at line 62 of file AVL.cpp.

**3.1.3.7** `void AVLTree::preOrder ( Node * pre ) const` [virtual]

Definition at line 183 of file AVL.cpp.

**3.1.3.8** `void AVLTree::print ( Node * node, int level )` [virtual]

Definition at line 198 of file AVL.cpp.

**3.1.3.9** `void AVLTree::printPreOrder ( ) const` [virtual]

Definition at line 177 of file AVL.cpp.

**3.1.3.10** `void AVLTree::push ( const int & n )` [virtual]

Definition at line 21 of file AVL.cpp.

**3.1.3.11** `bool AVLTree::search ( const int & s )` [virtual]

Definition at line 103 of file AVL.cpp.

**3.1.3.12** `void AVLTree::singleLeftRotate ( Node *& n )` [virtual]

Definition at line 39 of file AVL.cpp.

3.1.3.13 void AVLTree::singleRightRotate ( Node \*& n ) [virtual]

Definition at line 26 of file AVL.cpp.

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

- D:/STUDIA/PAMiSI/209226/AVL/[AVL.h](#)
- D:/STUDIA/PAMiSI/209226/AVL/[AVL.cpp](#)

## 3.2 Node Struct Reference

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

### Public Member Functions

- [Node](#) ()
- [Node](#) (const int &v, [Node](#) \*l, [Node](#) \*r, int h)

### Public Attributes

- [Node](#) \* left
- [Node](#) \* right
- int height
- int data

### 3.2.1 Detailed Description

Definition at line 7 of file node.h.

### 3.2.2 Constructor & Destructor Documentation

3.2.2.1 Node::Node ( ) [inline]

Definition at line 13 of file node.h.

3.2.2.2 Node::Node ( const int & v, Node \* l, Node \* r, int h ) [inline]

Definition at line 14 of file node.h.

### 3.2.3 Member Data Documentation

3.2.3.1 int Node::data

Definition at line 12 of file node.h.

3.2.3.2 int Node::height

Definition at line 11 of file node.h.



### 3.2.3.3 **Node\*** Node::left

Definition at line 9 of file node.h.

### 3.2.3.4 **Node\*** Node::right

Definition at line 10 of file node.h.

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

- D:/STUDIA/PAMiSI/209226/AVL/[node.h](#)

## Chapter 4

# File Documentation

### 4.1 D:/STUDIA/PAMiSI/209226/AVL/AVL.cpp File Reference

```
#include "AVL.h"
```

### 4.2 D:/STUDIA/PAMiSI/209226/AVL/AVL.h File Reference

```
#include <iostream>
#include <string>
#include "node.h"
```

#### Classes

- class [AVLTree](#)

### 4.3 D:/STUDIA/PAMiSI/209226/AVL/node.h File Reference

```
#include <stdlib.h>
```

#### Classes

- struct [Node](#)

### 4.4 D:/STUDIA/PAMiSI/209226/AVL/test/test\_AVL.cpp File Reference

```
#include <string>
#include <fstream>
#include <math.h>
#include "common/logger.h"
#include "common/catch.hpp"
#include "../AVL.h"
#include "common/observer.h"
```

## Macros

- `#define CATCH_CONFIG_MAIN`

## Functions

- `TEST_CASE` ("AVL binary tree, simple test", "[factorial]")

### 4.4.1 Macro Definition Documentation

#### 4.4.1.1 `#define CATCH_CONFIG_MAIN`

Definition at line 8 of file test\_AVL.cpp.

### 4.4.2 Function Documentation

#### 4.4.2.1 `TEST_CASE` ( "AVL binary *tree*, simple test" , "" *[factorial]* )

Definition at line 19 of file test\_AVL.cpp.