

Treap

Generated by Doxygen 1.8.20

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 Class Documentation	5
3.1 AbstractBST< KeyType, ValueType > Class Template Reference	5
3.1.1 Member Function Documentation	5
3.1.1.1 search()	5
3.2 TreapBST< KeyType, ValueType > Class Template Reference	6
3.2.1 Constructor & Destructor Documentation	6
3.2.1.1 TreapBST() [1/2]	6
3.2.1.2 TreapBST() [2/2]	6
3.2.1.3 ~TreapBST()	6
3.2.2 Member Function Documentation	7
3.2.2.1 height()	7
3.2.2.2 operator=()	7
3.2.2.3 search()	7
Index	9

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

AbstractBST< KeyType, ValueType >	5
TreapBST< KeyType, ValueType >	6

Chapter 2

Class Index

2.1 Class List

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

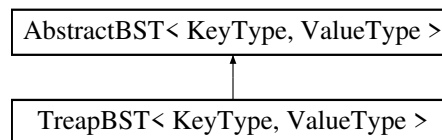
AbstractBST< KeyType, ValueType >	5
TreapBST< KeyType, ValueType >	6

Chapter 3

Class Documentation

3.1 AbstractBST< KeyType, ValueType > Class Template Reference

Inheritance diagram for AbstractBST< KeyType, ValueType >:



Public Member Functions

- virtual bool **empty** ()=0
- virtual ValueType **search** (const KeyType &key, bool &found)=0
- virtual void **insert** (const KeyType &key, const ValueType &value)=0
- virtual void **remove** (const KeyType &key)=0

3.1.1 Member Function Documentation

3.1.1.1 search()

```
template<typename KeyType , typename ValueType >
virtual ValueType AbstractBST< KeyType, ValueType >::search (
    const KeyType & key,
    bool & found ) [pure virtual]
```

Search for key. If found is true, returns the value associated with that key. If found is false, returns a default constructed ValueType.

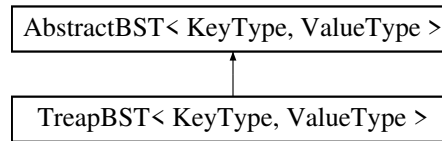
Implemented in [TreapBST< KeyType, ValueType >](#).

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

- abstract_bst.hpp

3.2 TreapBST< KeyType, ValueType > Class Template Reference

Inheritance diagram for TreapBST< KeyType, ValueType >:



Public Member Functions

- [TreapBST](#) ()
- [TreapBST](#) (const [TreapBST](#) &x)
- [TreapBST](#) & [operator=](#) ([TreapBST](#) x)
- [~TreapBST](#) ()
- bool **empty** ()
- ValueType [search](#) (const KeyType &key, bool &found)
- void **insert** (const KeyType &key, const ValueType &value)
- void **remove** (const KeyType &key)
- std::size_t [height](#) ()

3.2.1 Constructor & Destructor Documentation

3.2.1.1 TreapBST() [1/2]

```
template<typename KeyType , typename ValueType >
TreapBST< KeyType, ValueType >::TreapBST ( )
```

Default constructor.

3.2.1.2 TreapBST() [2/2]

```
template<typename KeyType , typename ValueType >
TreapBST< KeyType, ValueType >::TreapBST (
    const TreapBST< KeyType, ValueType > & x )
```

Copy constructor.

3.2.1.3 ~TreapBST()

```
template<typename KeyType , typename ValueType >
TreapBST< KeyType, ValueType >::~~TreapBST ( )
```

Destructor.

3.2.2 Member Function Documentation

3.2.2.1 height()

```
template<typename KeyType , typename ValueType >
std::size_t TreapBST< KeyType, ValueType >::height ( )
```

Get the height of the treap.

3.2.2.2 operator=()

```
template<typename KeyType , typename ValueType >
TreapBST& TreapBST< KeyType, ValueType >::operator= (
    TreapBST< KeyType, ValueType > x )
```

Copy-assignment.

3.2.2.3 search()

```
template<typename KeyType , typename ValueType >
ValueType TreapBST< KeyType, ValueType >::search (
    const KeyType & key,
    bool & found ) [virtual]
```

Search for key. If found is true, returns the value associated with that key. If found is false, returns a default constructed ValueType.

Implements [AbstractBST< KeyType, ValueType >](#).

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

- treap_bst.hpp

Index

`~TreapBST`
 [TreapBST< KeyType, ValueType >, 6](#)

`AbstractBST< KeyType, ValueType >`, [5](#)
 [search, 5](#)

`height`
 [TreapBST< KeyType, ValueType >, 7](#)

`operator=`
 [TreapBST< KeyType, ValueType >, 7](#)

`search`
 [AbstractBST< KeyType, ValueType >, 5](#)
 [TreapBST< KeyType, ValueType >, 7](#)

`TreapBST`
 [TreapBST< KeyType, ValueType >, 6](#)
`TreapBST< KeyType, ValueType >`, [6](#)
 [~TreapBST, 6](#)
 [height, 7](#)
 [operator=, 7](#)
 [search, 7](#)
 [TreapBST, 6](#)