Lab 06

Generated by Doxygen 1.8.11

Contents

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 files with brief descriptions:

Chapter 3

Class Documentation

3.1 ExprTree< DataType > Class Template Reference

`#include <ExpressionTree.h>`

Public Member Functions

- ExprTree ()
  default constructor for ExprTree class
- ExprTree (const ExprTree &source)
  default copy constructor for ExprTree class
- ExprTree & operator= (const ExprTree &source)
  overloaded operator for ExprTree class
- ~ExprTree ()
  default destructor for ExprTree class
- void build ()
  builds the tree with the use of build_Helper()
- void expression () const
  prints out the expressions in the tree with the use of expression_Helper()
- DataType evaluate () const throw (logic_error)
  prints out the total value of the equation stored in the tree with the use of evaluate_Helper()
- void clear ()
  deletes the entirety of the tree with the use of clear_Helper()
- void commute ()
  applies the commutative property with the use of the commute_Helper()
- bool isEquivalent (const ExprTree &source) const
- void showStructure () const

3.1.1 Constructor & Destructor Documentation

3.1.1.1 template<typename DataType > ExprTree< DataType >::ExprTree ( )

default constructor for ExprTree class

Precondition

none

Postcondition

sets root equal to null

Parameters

none	

Returns

none

3.1.1.2 template<typename DataType > ExprTree< DataType >::ExprTree (const ExprTree< DataType > & *source*)

default copy constructor for ExprTree class

Precondition

copyConstructor_Helper()

Postcondition

sets current tree equal to parameterized tree

Parameters

predefined	ExprTree

Returns

∗this, which is the current list whether its identical or needs to be rebuilt

3.1.1.3 template<typename DataType > ExprTree< DataType >::~ExprTree ( )

default destructor for ExprTree class

Precondition

clear()

Postcondition

destroys the current tree

Parameters

none	

Returns

none

3.1.2 Member Function Documentation

3.1.2.1 template<typename DataType > void ExprTree< DataType >::build ( )

builds the tree with the use of build_Helper()

Precondition

build_Helper()

Postcondition

builds an Expression Tree

Parameters

none	

Returns

none

3.1.2.2 template<typename DataType > void ExprTree< DataType >::clear ( )

deletes the entirety of the tree with the use of clear_Helper()

Precondition

clear_Helper()

Postcondition

destroys the current tree

Parameters

none	

Returns

none

3.1.2.3 template<typename DataType > void ExprTree< DataType >::commute ( )

applies the commutative property with the use of the commute_Helper()

Precondition

commmute_Helper()

Postcondition

applies the commutative property to the current expressions stored in the tree

Parameters

none	

Returns

none

3.1.2.4 template<typename DataType > DataType ExprTree< DataType >::evaluate ( ) const throw logic_error)

prints out the total value of the equation stored in the tree with the use of evaluate_Helper()

Precondition

evaluate_Helper()

Postcondition

none

Parameters

none	

Returns

evaluate_Helper(), using root as the start location

3.1.2.5 template<typename DataType > void ExprTree< DataType >::expression ( ) const

prints out the expressions in the tree with the use of expression_Helper()

Precondition

expression_Helper()

Postcondition

none

Parameters

none	

Returns

none

3.1.2.6 template<typename DataType > bool ExprTree< DataType >::isEquivalent (const ExprTree< DataType > & *source*) const

3.1.2.7 template<typename DataType > ExprTree< DataType > & ExprTree< DataType >::operator= (const ExprTree< DataType > & *source*)

overloaded operator for ExprTree class

Precondition

copyConstructor_Helper()

Postcondition

sets current tree equal to parameterized tree

Parameters

predefined	ExprTree

Returns

*this, which is the current list whether its identical or needs to be rebuilt

3.1.2.8 template<typename DataType > void ExprTree< DataType >::showStructure ( ) const

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

- ExpressionTree.h
- ExpressionTree.cpp

Chapter 4

File Documentation

4.1 ExpressionTree.cpp File Reference

An Implementation file for an Expression Tree.

```
#include "ExpressionTree.h"
```

4.1.1 Detailed Description

An Implementation file for an Expression Tree.

Author

Christopher Eichstedt

4.2 ExpressionTree.h File Reference

```
#include <stdexcept>
#include <iostream>
```

Classes

- class ExprTree< DataType >

Index