Group Eight Binary Tree Implemented in C
University of Montana Spring 2019 CSCI 205

Cole Brooks, Conner Copeland, Brooke Kern, Brendan Hagan

15-03-2019

1 Overview

This is a binary search tree implemented in C by Group 8 in the Spring 2019 section of CSCI 205 at the University of Montana. It supports reading space separated integers from a file specified at runtime through a command line argument. A full list and description of the available functions can be found in the "Function Documentation" section below. In addition to the functions supplied within the source code, the repository includes a makefile, and a bash script for testing the functionality of the program.

2 Function Documentation

- void initialize(binary_tree* bt);
  - Sets the size of the new tree to zero.
- void insert(binary_tree* bt, int item);
  - Inserts a new node containing the given item into the tree.
- bool search(binary_tree* bt, int key);
  - Returns whether given key is in the tree.
- void printinorder(binary_tree* bt);
  - Prints the tree contents from smallest node to largest.
- void printpreorder(binary_tree* bt);
  - Prints sequence of nodes derived from preorder traversal.
- void printpostorder(binary_tree* bt);
  - Prints sequence of nodes derived from postorder traversal.
- int btsize(binary_tree* bt);
  - Returns number of node in the tree.
- int treeheight(binary_tree* bt);
  - Returns height of the tree.