Xiao-Bao Bao

Period 4

Comp. Sci 3-4

G. Volger

Tree Program

**Overview:**

We needed to create a program that implemented functions related to the manipulation of Binary Trees. These functions included various traversals of the Binary Tree, displaying tree diagnostics such as the number of leaves, the height of the tree, the width of the tree, and whether or not a node exists on that tree. An important method that was implemented was one where a particular node had to be deleted on the tree while maintaining the structure of the tree. This was by far the most complex method; requiring the implementation of a couple helper methods for finding the smallest value in a Binary Tree as well as deleting the smallest value.

We needed to run the following instructions and provide the output:

* Load the first 20 integer values from [tree.txt](http://www.ljhs.sandi.net/faculty/volger/apabjava/TreeProgram/tree.txt)
* Draw the tree.
* Show the results for preorder, inorder, postorder levelorder traversals, as well the number of nodes, number of leaves, sum of all the noes in the tree, height and width of the tree.
* Search for the values: 555, 116 and 28. Display your results.
* Delete values 555, 116, 497, 961, 633, and 644 in that order, then do a preorder, inorder, and postorder traversal.
* Draw the tree.
* Destroy the tree, and do a preorder traversal.

Note that drawing a tree was *optional extra credit*. In our program ***drawing the Binary Tree has not been implemented***.

**Output:**

**Tree Functions:**

Now we will examine each of the functions that needed to be implemented as well as explain the functionality of the two helper methods that we implemented. We will go over the theory behind each function as well as how we implemented our solution for each function. (I will omit the explanation behind the *makeTree()* function as that method was implemented by the instructor and does not require further explanation)

*preorder()* –

*postorder()-*

*inorder()* –

*levelorder() –*

*numNodes()-*

*isLeaf()-*

*numLeaves() -*

*sumNodes() -*

*max() –*

*height() –*

*width() –*

*find() –*

*findMin() –*

*removeMin() –*

*deleteNode() –*

*destroy() –*