Isaiah Holder

Lab-8

Pointer implementation of a BST using recursion

**Operations Input Values Expected Output**

*InsertItem(int):*

Insert 10 values into the tree. 5,2,3,1,8,9,7,4,6,10 void

*Print(PRE\_ORDER,root):*

Print the values usine the

pre-order traversal. void 5,2,1,3,4,8,7,6,9,10

*Print(IN\_ORDER,root):*

Print the values using the

in-order traversal. void 1,2,3,4,5,6,7,8,9,10

*Print(POST\_ORDER,root):*

Print the values using the

post-order traversal. void 1,4,3,2,6,7,10,9,8

*RetrieveItem(&bool,int)-Positive Case:*

Search for a value known tobe

included in the original set of

numbers inserted into the structure. 23 Item Found

*RetrieveItem(&bool,int)-Negitive Case:*

Search for a value known tobe

absent from the original set

of numbers inserted into the structure. 230 Item Not Found

*DeleteItem()-No childeren:*

Delete a node with no childeren 6 1,2,3,4,5,7,8,9,10

*DeleteItem()-Right child:*

Delete a node with only one child

branching to the right. 9 1,2,3,4,5,7,8,10

*DeleteItem()-Two childeren/Left child:*

Delete a node with two childeren. 8 1,2,3,4,5,7,10

**Continue onto next page**

***Continued from page one:***

**Operations Input Values Expected Output**

*GetNextItem():*

The GetNextItem function is void void

tested within the print function.

*GetLength():*

Call GetLength void 7

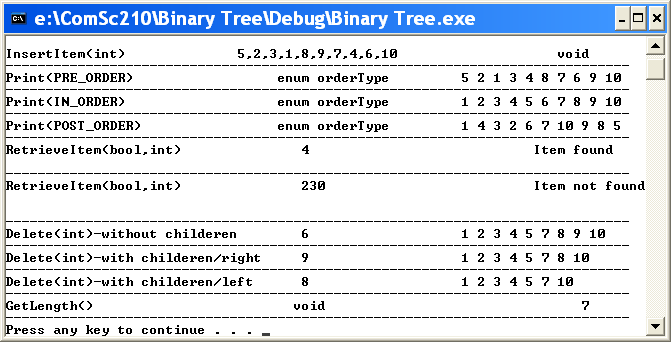
*MakeEmpty():*

Call MakeEmpty() void void

*ResetTree():*

Call ResetTree() void void

***Screen shot of operation:***

******