**Assignment 4**

**Due, Sunday, April 19, 2015 up to 100**

**Monday, April 20, 2015 up to 90**

**Tuesday, April 21, 2015 up to 80**

**Wednesday, April 22, 2015 up to 70**

**Beyond 4/22/2015, beg for mercy from Professor Whiting!**

**Deliverables**

To complete this assignment you must submit to Webcourses:

1. source code .c file

**Introduction**

This assignment provides practical experience with binary trees and binary search trees.

**References**

1. Text book
2. Source code examples on Webcourses,, Recursion.c, BinaryTree.c
3. Online tutorials:
   1. <http://www.cprogramming.com/tutorial/c-tutorial.html>
   2. <http://fresh2refresh.com/>
   3. <http://www.tutorialspoint.com/c_standard_library/>

**ATTENTION MAC USERS!**

If you are using a Mac operating system and Code::Blocks, I understand that you have to fully qualify the path to the files for the program to run. I ask that before you submit your assignment please remove the fully qualified path so that the **fopen("AssignmentFourInput.txt", "r");** function call has only the file name. Thank you!

If you are using a Mac operating system and Xcode, place your .txt files in the directory Products, where the .exe is written to you don’t have to use a fully qualified path, just use the file name.

**Tasks and Rubric**

|  |  |  |
| --- | --- | --- |
| Activity | | Points |
| Define a binary search tree node to include | struct binTreeNode  {  int data;  struct binTreeNode \* right;  struct binTreeNode \* left;  }; | 5 |
| main() | - Provide the user a menu of the following options:  1. Generate Binary Search Tree  2. Print the BST in pre-order format  3. Print the BST in in-order format  4. Print the BST in post-order format  5. Print the BST in breadth-first format  6. Find a value in the BST  7. Find the minimum value in the BST nodes  8. Find the maximum value in the BST nodes  9. Calculate the average value of the BST nodes  10. Find the median value of the BST nodes  11. Calculate the sum of the BST nodes  12. Count the number of BST nodes  13. Delete a value in the BST  14. Exit Program  - Use a conditional statement to evaluate the user’s selection | 5 |
| Function to read the data file as nodes of a binary search tree | Read in the contents of data file "AssignmentFourInput.txt" into the node data structure and generate a binary search tree in the order the data is in the file | 5 |
| Function to display the BST in pre-order format | Traverse the BST and display the nodes in pre-order format | 5 |
| Test Case 1 | Perform Test Case 1, results in expected outcome | 5 |
| Function to display the BST in in-order format | Traverse the BST and display the nodes in in-order format | 5 |
| Test Case 2 | Perform Test Case 2, results in expected outcome | 5 |
| Function to display the BST in post-order format | Traverse the BST and display the nodes in post-order format | 5 |
| Test Case 3 | Perform Test Case 3, results in expected outcome | 5 |
| Function to display the BST in breadth-first format | Traverse the BST and display the nodes in breadth-first order |  |
| Test Case 4 | Perform Test Case 4, results in expected outcome | 5 |
| Function to find a value in the BST | Traverse the BST and search for the value the user entered, display to the user the result of their search (i.e. if the value was found or not found) |  |
| Test Case 5 | Perform Test Case 5, results in expected outcome | 5 |
| Test Case 6 | Perform Test Case 6, results in expected outcome | 5 |
| Function to find the minimum value in the BST | Traverse the BST and search for the minimum value in the BST |  |
| Test Case 7 | Perform Test Case 7, results in expected outcome | 5 |
| Function to find the maximum value in the BST | Traverse the BST and search for the maximumvalue in the BST |  |
| Test Case 8 | Perform Test Case 8, results in expected outcome | 5 |
| Function to find the average value of the nodes in the BST | Traverse the BST and calculate the average value of the BST nodes |  |
| Test Case 9 | Perform Test Case 9, results in expected outcome | 5 |
| Function to find the median value of the nodes in the BST | Traverse the BST and determine the median value of the BST nodes |  |
| Test Case 10 | Perform Test Case 10, results in expected outcome | 5 |
| Function to find the sum of the values of the BST nodes | Traverse the BST and determine the sum of the values of the BST nodes |  |
| Test Case 11 | Perform Test Case 11, results in expected outcome | 5 |
| Function to find the count of the of the BST nodes | Traverse the BST and find the count of the BST nodes |  |
| Test Case 12 | Perform Test Case 12, results in expected outcome | 5 |
| Function to delete a node in the BST | Search the BST for the entered value and delete it |  |
| Test Case 13 | Perform Test Case 13, results in expected outcome | 5 |
| Exit the program | Write the appropriate code to exit the program | 5 |
| Test Case 14 | Perform Test Case 14, results in expected outcome | 5 |
| Compile | Source compiles with no errors | 5 |
| Run | Source runs with no errors | 5 |
| Comments | Source includes comments | 5 |
| Total |  | **100** |

**Perform the following test cases**

|  |  |  |
| --- | --- | --- |
| Test Cases | | |
|  | **Action** | **Expected outcome** |
| Test case 1 | Select option 1 then option 2 | Display the elements of the BST in pre-order format |
| Test case 2 | Select option 3 | Display the elements of the BST in in-order format |
| Test case 3 | Select option 4 | Display the elements of the BST in post-order format |
| Test case 4 | Select option 5 | Display the elements of the BST in breadth-first format |
| Test case 5 | Select option 6 | Search the BST for a value that exists |
| Test case 6 | Select option 6 | Search the BST for a value that does not exist |
| Test case 7 | Select option 7 | Display the minimum value of the BST |
| Test case 8 | Select option 8 | Display the maximum value of the BST |
| Test case 9 | Select option 9 | Display the average value of the BST |
| Test case 10 | Select option 10 | Display the median value of the BST |
| Test case 11 | Select option 11 | Display the sum of the BST nodes |
| Test case 12 | Select option 12 | Display the count of the BST nodes |
| Test case 13 | Select option 13 | Delete a node with a value that exists in the BST |
| Test case 13 | Select option 13 | Delete a node with a value that does not exist in the BST |
| Test case 14 | Select option 14 | Program exits |