DESCRIPTION

Donation Trees

Goals

- Implement a BST
- Use the BST to solve a problem

Matthew's Stats

Time Taken: 5 hours

Part 1

In the starter code, you have been provided with the definition of a class for a Binary Search Tree as well as a BSTNode. Implement the methods that you find within it. If you are confused about what a particular method is supposed to do or what a value means please post it on CampusWire

While you must write finish at least the functions provided you are free to add more functions if you would like. I certainly did in my solution.

Unit Testing

You need to write sensible unit tests using Python's unittest module

You must test **at least** the following methods of your tree class

- __ init __
- get_min_node
- get_max_node
- add_value
- remove_value
- height
- __ len __

To help you test I have provided you with the definition for == between two trees. I have also defined __ iter __ for your BSTNode class as BST.__ eq __ depends on it to work correctly. No need to add any code to these functions.

We will be looking over your test cases to see if they make sense and test your code thoroughly. Please do your best to make this section of code as readable as possible because if we can't understand it we can't give you credit.

To make sure that your testing is thorough

- Test a BST of strings ordered by the their ASCII code value
- Test a BST of Strings where the elements are ordered by their length

Doing this should help you catch more mistakes in your code than if you only ever create a BST of numbers.

Mimir Testing

I will be running my own set of unit tests on your BST to verify that it is working correctly.

Locations and Names of Files and Folders

You are free to add more files and directories to your project but you cannot more or rename any of the files in the starter code.