CSCI B505 Fall 19: Programming assignment 5

Due online: Friday, November 29, 11:59pm EST

Submit your work online using Canvas. You can use LaTeX, Word, or even pen and paper for the write-up. But please try to submit a PDF file. Use the lab sessions prior to the due date to get help from your AI/UI, if necessary.

What to submit

For this assignment you will be submitting two things:

1. Source code. Please, follow good coding practices: use indentation, write comments, etc.
2. Write-up. This should contain:
   - Description of how you solved both problems, including the two printed outputs for Part 2.
   - Justifications for any choices you've made.
   - Conclusions and analysis of the results.

Part 1

Implement a Binary Search Tree data structure as a class which supports the following queries. You can assume that all keys are integer numbers and there are *no duplicates*.

- **insert(key)** – inserts *key* into the tree.
- **contains(key)** – returns True if *key* is in the tree, otherwise False.
- **inorder()** – prints elements of the tree inorder.
- **size()** – returns the number of nodes in the tree.
- **smallest()** – returns the smallest element in the tree.
- **largest()** – returns the largest element in the tree.
- **successor(key)** – returns the smallest element in the tree whose value is greater than *key*.
- **predecessor(key)** – returns the largest element in the tree whose value is less than *key*.

Discuss the worst-case time and space complexities of each of the queries.

Part 2

Using the Binary Search Tree class, implement a function GREATERSUMTREE() which transforms a given tree so that each node contains the sum of all other keys in the tree which are greater than this node. Analyze the time and space complexity of this function. In addition, call INORDER() on the tree before and after you perform GREATERSUMTREE() and include the output into the discussion.

Hints

- Chapter 12 and the slides contain pseudocode for most of the functions above. You can use them.
- Ask questions on Piazza.