

CS 2420-001 ALGORITHMS AND DATA STRUCTURES

Spring Semester, 2014

Assignment 3: Binary Search Trees

Due Date: Friday, Feb. 14, 2014 (at the beginning of CS2420 class)

(**Note:** This assignment has only one programming exercise. Please submit your cpp file to Canvas.)

1. In this exercise, we will implement the following operations of binary search trees we discussed in class. **(60 points)**
 - (a) *insert*(x): insert a new node whose key is x to the tree. If there is already a node whose key is x , then we do nothing.
 - (b) *remove*(x): remove the node whose key is x from the tree. If x is not in the tree, then we do nothing. As discussed in class, if the node x has two children, then please use the in-order successor of x to replace the node x .
 - (c) *search*(x): determine whether the key x is in the tree. If yes, return “true”; otherwise return “false”.
 - (d) *findMin*(): return the smallest key of the tree.

On Canvas, go to the following directory: homework/hw3/question1. There are a starter cpp file “hw3_Q1.cpp” and an input file “hw3.Q1.input.txt”. Each line of the input file is one of the following types: “insert x”, “remove x”, “search x”, or “findMin”. The program reads the input file line by line and perform the operations accordingly. For each search or findMin operation, the result will be stored in an output file “hw3_Q1_output.txt”. Finally, the program will print both pre-order and in-order traversal lists of the tree to the output file (the traversal functions have already been provided).

In addition, in order to help you to check whether your program runs correctly, I put my file “Wang_hw3_Q1_output.txt” in the same directory, which contains the correct output for the data in the input file. I got the file using my own program. Again, if you want to take a look at the file in Windows system, please use “WordPad” instead of “Notepad” to open it.

Your task is to complete the four functions: *insert*(), *remove*(), *search*(), and *findMin*(). For each function, you may write the code using either the iterative or recursive approach we discussed in class. You can also overload these functions, as I did for the traversal functions in the starter cpp file or as we discussed in class.

Again, please follow the instructions we used in Assignment 1, e.g., submit a single cpp file to Canvas; change the file name by adding your A number at the end; do not change the input/output style in your submitted program.

Total Points: 60