

COP 3502 Computer Science I Spring 2016
Assignment 6
Due Sunday 10th April at 11:30 pm.
This assignment is worth a maximum of 100 points

Objective

By completing this assignment, students will demonstrate a working knowledge of the following Binary Search Tree (BST) concepts:

1. Code implementation of BST
2. Traversing BST
3. Algorithm for inserting nodes into a binary search tree
4. Deleting nodes from a BST
5. Searching for values in BST

Problem: Manipulating Real Estate Database

You are required to create software to manage a real estate database. Because the system must be frequently searched and updated, your team-lead has suggested that you use a BST, ordered on the listingID, to store all of the property listings.

Each property listing is comprised of the following information:

- Listing ID (an integer)
- Agent (a string no greater than 12 characters long with no spaces)
- Price (an integer)
- Size in square footage (an integer)
- Number of beds (an integer)
- Number of baths (a double)
- Year built (an integer)
- Address (a string, no longer than 30 characters long)

Your code should be able to handle the following database operations:

- 1) Adding a property listing
- 2) Deleting a listed property
- 3) Searching for a listed property by ID
- 4) Searching for properties listed up to a specified price
- 5) Searching for listed properties by Agent
- 6) Listing all of the available properties

When run, your program should read real estate information from an input file called *listings.txt*. It should store this information in a database implemented as a BST, built as each listing is read. Your program should then execute a given set of operations on the created database.

Input Specification (for listings.txt)

The first line of the file will contain a single positive integer n representing the number of properties listed. This will then be followed by n lines of data; each representing a single property. Each line of data will begin with an integer representing the listingID, a string representing the agent, an integer representing the cost of the property. This is followed by: the property size (an integer)
Number of beds (an integer)
Number of baths (a double)
Year built (an integer)
Address (a string, no longer than 30 characters long)

The rest of the file is a series of lines each with a database operation to be executed. Each line begins with an integer from 1 – 6 inclusive, indicating the database operation to be executed. The numbers correspond to the list on the previous page.

If the operation is (1), to add a property to the listings, then the line will contain the all of the listing information, in the order given above.

If the operation is (2), to delete a listing, then the line will contain one other integer representing the listingID of the property to be deleted.

If the operation is (3), to search for a listing by ID, then the line will contain one other integer representing the listingID of the property to be searched for.

If the operation is (4), to search for a listing by price, then the line will contain an integer representing the maximum price of the properties to be searched for.

If the operation is (5), to list all of properties under a given agent, then the line will contain the agent's name after the operation number.

If the operation is (6), to print all of the available properties, then that will be the only number on the line

The input file ends with a 0.

Output Specification

Your output should match that in the sample output. You are to use the exact text for the functions as shown.

Implementation Restrictions

- a. `#define ADD_LENGTH 30` (for address length)
- b. Each Listing must be implemented as a struct
- c. Each BST node should have a listing as its data
- d. Your program should be written using appropriate functions.
- e. Do not use global variables. Instead pass parameters to your functions. Remember that for functions that will be changing your BST, it (the BST), should be passed as a parameter by reference.
- f. Deletions of listings should be implemented using the successor algorithm

Deliverables

Your assignment should be submitted via Webcourses by the deadline. No assignments will be accepted via email. Your submission should include a single .c file called **realEstateDB.c**. Remember that your program will be graded using Eustis, so be sure that your submitted code runs correctly on the server.

Remember that it is each student's responsibility to ensure that the correct file has been uploaded. Thus, as advised at the beginning of the semester, immediately after submitting your code, download it and verify that you have submitted the intended file and that the submission has gone through. An excuse that there was an upload error is not grounds for late submission. No assignments will be accepted after the 24 hour window for late submissions via Webcourses.

Sample Input File (library.txt)

6

121 Matt 205000 3000 3 2.0 2000 35 Blueberry Lane
163 Amy 450000 5000 5 5.5 2016 8885 Winter Garden Drive
116 Grant 375000 3000 3 2.0 2015 191 Oviedo Lakes
100 Linda 355000 4000 4 2.5 2014 79 Bradmore Lane
155 Amy 495000 3500 3 3.0 2012 52 Lenox Drive
280 Grant 950000 5000 5 6.5 2016 223 Willow Pines

6

1 119 Linda 985000 2950 2 2.0 2013 1200 Park Avenue
2 116
3 132
5 Amy
4 400000
6
0

Sample Output (corresponding to input file)

LID	Agent	Price \$K	size	Bd	Bth	Yr	Address
100	Linda	355	4000	4	2.5	2014	79 Bradmore Lane
116	Grant	375	3000	3	2.0	2015	191 Oviedo Lakes
121	Matt	205	3000	3	2.0	2000	35 Blueberry Lane
155	Amy	495	3500	3	3.0	2012	52 Lenox Drive
163	Amy	450	5000	5	5.5	2016	8885 Winter Garden Drive
280	Grant	950	5000	5	6.5	2016	223 Willow Pines

Listing 119 has been added to the database

Listing 116 has been deleted from the database

Listing 132 does not exist in the database

LID	Agent	Price \$K	size	Bd	Bth	Yr	Address
155	Amy	495	3500	3	3.0	2012	52 Lenox Drive
163	Amy	450	5000	5	5.5	2016	8885 Winter Garden Drive

LID	Agent	Price \$K	size	Bd	Bth	Yr	Address
100	Linda	355	4000	4	2.5	2014	79 Bradmore Lane
116	Grant	375	3000	3	2.0	2015	191 Oviedo Lakes
121	Matt	205	3000	3	2.0	2000	35 Blueberry Lane

LID	Agent	Price \$K	size	Bd	Bth	Yr	Address
100	Linda	355	4000	4	2.5	2014	79 Bradmore Lane
119	Linda	985	2950	2	2.0	2013	1200 Park Avenue
121	Matt	205	3000	3	2.0	2000	35 Blueberry Lane
155	Amy	495	3500	3	3.0	2012	52 Lenox Drive
163	Amy	450	5000	5	5.5	2016	8885 Winter Garden Drive
280	Grant	950	5000	5	6.5	2016	223 Willow Pines