COP3530 Project 4 – Binary Search Trees

Due Date: Friday, 11/19/2021 11:59 PM

Turn in:

Submit the zipped Eclipse program including at least Project4.java, BinarySearchTree.java, Node.java and States4.csv. The zip file should be named <your last name>_Project4.zip (for example, Liu_Project4.zip). The States4.csv file contains information about all 50 States in the USA. For each State, in the CSV are its name, capitol, region, US House seats and population (according to the *2020 Census* found in *Wikipedia* at https://en.wikipedia.org/wiki/List_of_states_and_territories_of_the_United_States_by_population), numbers of COVID-19 total cases and deaths (as of Sept 2, 2021, according to *Centers for Disease Control and Prevention* at https://covid.cdc.gov/covid-data-tracker/), median household income (according to the *2020 World Population Review* at *https://worldpopulationreview.com/state-rankings/median-household-income-by-state*), and violent crime rates (according to the *2019 FBI report* at https://ucr.fbi.gov/crime-in-the-u.s/2019/crime-in-the-u.s.-2019, did not find 2020 FBI data for this).

The program should be well documented in the format of doc comments in Java. Detailed formats are found at http://www.oracle.com/technetwork/articles/java/index-137868.html.

Requirements:

(1) . Create a **Node** class, where you will have as data members the state name, the death rate of the state (2 decimal precisions), the left child reference, and the right child reference, and you will have a print method to print the node according to some format.

(2) .Create the **BinarySearchTree** class which should implement the following public methods:

1. A no-parameter constructor that creates an empty tree.
2. The method: **public void insert(String name, double DR)** that will insert a node into the proper position in the search tree based on state name.
3. The method: **public double find(String name)** that will search the tree for the state of the given name and if found will return the DR or -1 if not found. If found, this method should print out the ***path*** of the found node. The path of a node is defined as the sequence of nodes from root to the node.
4. The method: **public void delete(String name)** that will find and delete the given state from the tree.
5. The method: **public void printInorder()** that will traverse the tree in using a Inorder traversal (LNR) and print each node.

6. The method: **public void printPreorder()** that will traverse the tree in using a Preorder traversal (NLR) and print each node.

7. The method: **public void printPostorder()** that will traverse the tree in using a Postorder traversal (LRN) and print each node.

8. The method: **public void printBottomStates(int c)** that will find and print in order the bottom c states regarding *DR*, that is, the c states with highest DR. These states should be printed in a descending order. If there are less than c states in the tree, print all of them. (Note: for this method, you are ***required NOT to use any extra binary search tree and NOT to use any sorting method. Hint: you may consider using an array of size c and a few other constant-memory variables***.)

9. The method: **public void printTopStates(int c)** that will find and print in order the top c states regarding *DR*, that is, the c states with lowest DR. These states should be printed in an ascending order. If there are less than c states in the tree, print all of them. (Note: for this method, you are ***required NOT to use any extra binary search tree and NOT to use any sorting method. Hint: you may consider using an array of size c and a few other constant-memory variables***.)

(3) . Create a class called **Project4** that will

1. Prompt user to enter the name of the CSV file, e.g., States4.csv, as input to the system

2. Read the CSV file to create a binary search tree by calling the **insert** method. (This should be the **ONLY** tree maintained throughout the program.)

3. Offer the user the following functionalities:

   1) Print tree inorder
   2) Print tree preorder
   3) Print tree postorder
   4) Delete a state for a given name
   5) Search and print a state and its path for a given name.
   6) Print bottom states regarding DR
   7) Print top states regarding DR
   8) Exit

Provide comments in this form for the **BinarySearchTree** classes:

Comments for the class:

```
/**
 * Detailed description of the class.
 *
 * @author <your name>
 * @version <date you last changed the class>
 */
```

Public method comments:

```
/**
 * Description of the purpose of the method, the meaning of the
 * input parameters (if any) and the meaning of the return values * (if any).
 *
 * @param  parameter  description of the parameter (one for each)
 * @return description of the return value
 */
```

Provide comments in this form for the **Project4** class.

```
/**
 * COP 3530: Project 4 – Binary Search Trees
 * <p>
 * Description of the class using as many lines as needed
 * with <p> between paragraphs. Including descriptions of the
 * input required and output generated.
 *
 * @author <your name>
 * @version <the date you last modified the program>
 */
public class Project4
{
```

Example Output:

```
COP3530 Project 4
Instructor: Xudong Liu

Enter the file name: States4.csv

There were 50 records read to build a binary search tree.

1) Print tree inorder
2) Print tree preorder
3) Print tree postorder
4) Delete a state for a given name
5) Search and print a state and its path for a given name
6) Print bottom states regarding DR
7) Print top states regarding DR
8) Exit
Enter your choice: 1

Inorder traversal:
Name          Death Rate
------------------------
Alabama       245.41
. . .


1) Print tree inorder
2) Print tree preorder
3) Print tree postorder
4) Delete a state for a given name
5) Search and print a state and its path for a given name
6) Print bottom states regarding DR
7) Print top states regarding DR
8) Exit
Enter your choice: 2

Preorder traversal:
Name          Death Rate
------------------------
Florida       213.15
. . .


1) Print tree inorder
2) Print tree preorder
3) Print tree postorder
4) Delete a state for a given name
5) Search and print a state and its path for a given name
6) Print bottom states regarding DR
7) Print top states regarding DR
8) Exit
Enter your choice: 3

Postorder traversal:
Name          Death Rate
------------------------
```

```
Alaska       59.31
. . .



1) Print tree inorder
2) Print tree preorder
3) Print tree postorder
4) Delete a state for a given name
5) Search and print a state and its path for a given name
6) Print bottom states regarding DR
7) Print top states regarding DR
8) Exit
Enter your choice: 4
Enter state name: Florida

Florida is deleted from binary search tree.



1) Print tree inorder
2) Print tree preorder
3) Print tree postorder
4) Delete a state for a given name
5) Search and print a state and its path for a given name
6) Print bottom states regarding DR
7) Print top states regarding DR
8) Exit
Enter your choice: 2

Preorder traversal:
Name          Death Rate
----------------------
Georgia       213.50
. . .



1) Print tree inorder
2) Print tree preorder
3) Print tree postorder
4) Delete a state for a given name
5) Search and print a state and its path for a given name
6) Print bottom states regarding DR
7) Print top states regarding DR
8) Exit
Enter your choice: 5
Enter state name: Washington

Washington is found with death rate of 85.80.
Path to Washington is Georgia -> … -> Washington



1) Print tree inorder
2) Print tree preorder
3) Print tree postorder
4) Delete a state for a given name
5) Search and print a state and its path for a given name
6) Print bottom states regarding DR
```

```
7) Print top states regarding DR
8) Exit
Enter your choice: 5
Enter state name: Florida

Florida is not found.



1) Print tree inorder
2) Print tree preorder
3) Print tree postorder
4) Delete a state for a given name
5) Search and print a state and its path for a given name
6) Print bottom states regarding DR
7) Print top states regarding DR
8) Exit
Enter your choice: 6
Enter the number of states: 5

Bottom 5 states regarding DR:
Name          Death Rate
-------------------------
New Jersey    289.61
. . .



1) Print tree inorder
2) Print tree preorder
3) Print tree postorder
4) Delete a state for a given name
5) Search and print a state and its path for a given name
6) Print bottom states regarding DR
7) Print top states regarding DR
8) Exit
Enter your choice: 7
Enter the number of states: 5

Top 5 states regarding DR:
Name          Death Rate
-------------------------
Hawaii        41.09
. . .



1) Print tree inorder
2) Print tree preorder
3) Print tree postorder
4) Delete a state for a given name
5) Search and print a state and its path for a given name
6) Print bottom states regarding DR
7) Print top states regarding DR
8) Exit
Enter your choice: 87
Invalid choice enter 1-8: 0
Invalid choice enter 1-8: A
Invalid choice enter 1-8: 8

Have a good day!
```