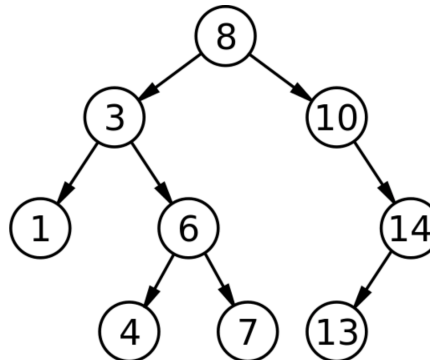


## CSCD 320 Homework Two –first choice

This is a programming assignment. Javadoc Standard Documentation **\*NOT\*** Required -- basic documentation is (all source files must include author name and a description of the class(es) contained in the file; any code another developer might not understand at first look should also be documented). Also, follow Java's naming conventions, utilize whitespace/indentation liberally in your code.

### Requirements

- 1) As we learned in class, please implement a Binary Search Tree(BST) and its nested tree Node class, as well as their constructors. Data type in the tree Node is Object.
- 2) As we learned in class, please implement the insert() method that performs insertion into the tree.
- 3) As we learned in class, please implement post-orderTraversal and in-orderTraversal method for the tree.
- 4) As we learned in class, please implement a method boolean delete( Object toRemove) in your BST class. The delete() method will search the tree and delete the Object toRemove you passed in as parameter, then return true. After deletion this tree maintains its BST properties. If the Object is not found in the tree, the method returns false. You are allowed to implement other helper methods for this delete method as needed.
- 5) Implement a Tester class to test your BST implementation. In the Tester class, instantiate a BST object myBST, then using insert() method to insert data items into myBST so that it will contain the data shown as below.



Hint: thinking about in what order you will insert the sequence of these integers to form this tree in the main memory?

- 6)After you created completely myBST object following step 5, you call your in-orderTraversal() method to traverse the tree. You are supposed to see the results are displayed on the standard output {1, 3, 4, 6, 7, 8, 10, 13, 14}, if your implementation is correct.

7) Following step 6, you call your `post-orderTraversal()` method to traverse the tree. You are supposed to see the results are displayed on the standard output `{1, 4, 7, 6, 3, 13, 14, 10, 8}`, if your implementation is correct.

8) In your `Tester` class, continuing the previous step, you call `myBST.delete( new Integer(8) )`, which attempts to delete the root node.

9) After step 8, call `in-orderTraversal()` method and `post-orderTraversal()` method again to verify deletion works properly. At this point, you are supposed to see the in-order traversal result `{1, 3, 4, 6, 7, 10, 13, 14 }`. And you are supposed to see the post-order traversal result `{1, 4, 6, 3, 13, 14, 10, 7}`.

10) Continuing on step 9, you call `myBST.delete(new Integer(10))`.

11) After step 10, call `in-orderTraversal()` method and `post-orderTraversal()` method again to verify deletion works properly. At this point, you are supposed to see the in-order traversal result `{1, 3, 4, 6, 7, 13, 14 }`. And you are supposed to see the post-order traversal result `{1, 4, 6, 3, 13, 14, 7}`.

12) At the bottom of your `Tester.java` file, please use java comments to briefly explain why you can get the post-order traversal results for step 11? Hint: explain as if you are the computer that executes the code step by step?

Java inline comment format is shown in the below, each line starting with double slashes.

```
// this is a java code comments
```

```
// This is explains blab ..
```

```
//.....
```

### To Turn In:

Turn in your solution on the **EWU Canvas** by going to CSCD320-01 course page on Canvas, then clicking Assignments→hw2->submit.

This assignment must be submitted in working order by the due data on the top of this assignment. Submit a zip file with your source files only. Source files are those that end in `.java`. Check your zip file before submitting and make sure it has only your source files. Do not submit `.class` files; they do the grader no good. You will not receive credit if your submission contains only `.class` files. Name your zip file with your last name, followed by the first initial of your first name, followed by `hw2`. For example, if you are John Smith, name you file as `smithjhw2.zip`.

The grader should be able to open your zip file, compile your code, and run your program from **the command line using `javac` and `java`** version 1.6 or greater (so make sure you try this yourself before you submit).

Get started right away on this assignment or you WILL NOT finish :-(

**If your code has compile-time error when USING COMMAND LINE tool to compile, you get a ZERO for this homework! That is, we use the command `javac *.java` to compile your source code and use `java Tester` to run your program.**