

APS 105 — Computer Fundamentals

Laboratory 8

Winter, 2001

To be submitted by noon on Tuesday, April 10

Objective

In this lab, you are required to use binary trees.

The Problem

This problem is very similar to the one seen in Lab 7 but the structures used to store the data have been changed. In this lab, you will be dealing with binary search trees defined, initially, by the following class.

```
class Tree
{
    private Node root;

    class Node
    {
        String info;
        Node lLink;
        Node rLink;
    }
}
```

Add (at least) the following methods to these classes.

- `public Tree ()`
A constructor for the `Tree` class that creates a new, empty tree
- `public Node (String word)`
A constructor for the `Node` class that creates a new `Node` object with `word` in its `info` field and `null` in its `lLink` and `rLink` fields
- `public void printTree ()`
A method that prints all the nodes of a binary tree using an inorder traversal
- `public void insert (String s)`
A method to insert `s` in a binary search tree
- `public void convertToLowercase ()`
A method to convert all letters of all strings in a tree to lowercase

Test your methods by writing, in a separate file called `TreeTest.java`, a program containing a `main` method that performs the following actions

- Repeatedly prompt the user to supply words until the user enters `ZZZ`. Each string given by the user (except for `ZZZ`) should be inserted in one of two binary search trees — one for words whose first character is an upper case letter and the other for words whose first character is a lower case letter. Assume that all strings provided by the user are single words. If a word is already in a tree, do not insert it a second time. Ignore the case of letters when comparing words. For example, if the word `SAMPLE` is already in a tree, the word `Sample` should *not* be inserted.
- Once all the input has been read, print the contents of both trees, in alphabetical order.

- Change all uppercase letters of the words to lowercase letters.
- Print the altered trees.
- Print any words that appear in both trees. You should write methods of your own design (for the `Tree` and `Node` classes) to perform this task.

What to Submit

Submit your `Tree.java`, `TreeTest.java`, and `Stdin.java` files using the command
`submitaps105s 8 List.java ListTest.java Stdin.java`