Project 0: Tree Assignment

**Due Date:** Tuesday, October 4, 2005 (Duration: One week)

This assignment is not for credit; do not hand anything in.

**Objectives**

Gain familiarity with the Java Language. Get used to working with tree-shaped data structures, which we’ll used to represent source programs in our compiler.

**Requirements**

Obtain the file `Tree.java`. It is available through the web page, or at:

```
~harry/public_html/compilers/p0
```

Make the following modifications to the code, and test your changes.

1. Add a method
   ```java
   String postFix ()
   ```
   to each of the classes. When invoked on a node in the tree, this method will return a String giving the post-fix representation of the subtree at that node. For example, the output might be:
   ```
   Prefix = * - + d e + g h
   Infix = ((- (d + e)) * (g + h))
   Postfix = d e + - g h + *
   ```

2. Add a method to walk the tree and return the number of nodes in the tree. Print this number out:
   ```
   Number of nodes = 8
   ```

3. Add a new kind of node with 3 children, called a ternary node. Such a node will have 2 labels. For example:
   ```java
   Node x = new TernaryNode (?", ":", a, b, c);
   ```
   A ternary node will look like this when printed:
   ```
   Prefix = ?: a b c
   Infix = (a ? b : c)
   Postfix = a b c ?:
   ```

**Additional Requirements:**

1. Add / modify comments as appropriate.
2. Follow the existing stylistic conventions in the code. Make your code look the same.
3. Always add your name and the date to any code files you modify.