```
1 /****
 2 *
 3 * TreeNode1 extends TreeNode by adding one child component, which is a
 4 * reference to another TreeNode. Hence, TreeNodel is used to represent unary
 5 * syntactic constructs in a parse tree.
 6 *
7 */
 8 public class TreeNodel extends TreeNode {
10
11
        * Construct this with the given id and child TreeNode reference.
12
13
       public TreeNode1(int id, TreeNode child) {
14
           super(id);
15
           this.child = child;
16
17
18
19
        * A la the other constructor, but with line and column numbers.
20
21
       public TreeNodel(int id, TreeNode child, int line, int column) {
22
            super(id, line, column);
            this.child = child;
23
24
25
26
27
        * Return the String representation of this subtree, which is the String
28
         * value of its ID, followed on the next indented line by the recursive
29
         * toString of its child. See the documentation for <a href=
         * "TreeNode.html#toString()"> TreeNode.toString() </a> for a general
30
        * description the way trees are represented as strings.
31
32
33
       public String toString(int level) {
34
           String indent = "";
           for (int i = 0; i < level; i++) {
35
36
              indent += " ";
37
38
           return symPrint(id) + toStringLineAndColumn(" ") + "\n" +
39
              indent + " " +
                   (child == null ? "null" : child.toString(level+1)) + "\n";
40
41
42
43
        /** Reference to the single child of this node. */
44
       public TreeNode child;
45
46 }
```