```
 1  /****
 2   *
 3   * TreeNode1 extends TreeNode by adding one child component, which is a
 4   * reference to another TreeNode.  Hence, TreeNode1 is used to represent unary
 5   * syntactic constructs in a parse tree.
 6   *
 7   */
 8  public class TreeNode1 extends TreeNode {
 9
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       * Return the String representation of this subtree, which is the String
20       * value of its ID, followed on the next indented line by the recursive
21       * toString of its child.  See the documentation for <a href=
22       * "TreeNode.html#toString()"> TreeNode.toString() </a> for a general
23       * description the way trees are represented as strings.
24       */
25      public String toString(int level) {
26          String indent = "";
27          for (int i = 0; i < level; i++) {
28              indent += "  ";
29          }
30          return symPrint(id) + "\n" +
31              indent + "  " + (child == null ? "null" : child.toString(level+1)) + "\n";
32      }
33
34      /** Reference to the single child of this node. */
35      public TreeNode child;
36
37  }
```