```
12    public int getAndIncrement() {
13      Stack<Node> stack = new Stack<Node>();
14      Node myLeaf = leaf[ThreadID.get()/2];
15      Node node = myLeaf;
16      // precombining phase
17      while (node.precombine()) {
18        node = node.parent;
19      }
20      Node stop = node;
21      // combining phase
22      int combined = 1;
23      for (node = myLeaf; node != stop; node = node.parent) {
24        combined = node.combine(combined);
25        stack.push(node);
26      }
27      // operation phase
28      int prior = stop.op(combined);
29      // distribution phase
30      while (!stack.empty()) {
31        node = stack.pop();
32        node.distribute(prior);
33      }
34      return prior;
35    }
```

FIGURE 12.4 The CombiningTree class: the getAndIncrement() method.