Worksheet #20: Sequential->Parallel Spanning Tree Algorithm

1. Insert finish, async, and isolated constructs (pseudocode is fine) to convert the sequential spanning tree algorithm from the other side below into a parallel algorithm.

2. Is it better to use a global isolated or an object-based isolated construct for the parallelization in question 1? If object-based is better, which object(s) should be included in the isolated list?
Worksheet #20: Sequential->Parallel Spanning Tree Algorithm

1. class V {
2. V [] neighbors; // adjacency list for input graph
3. V parent; // output value of parent in spanning tree
4. boolean makeParent(V n) {
5. if (parent == null) { parent = n; return true; }
6. else return false; // return true if n became parent
7. } // makeParent
8. void compute() {
9. for (int i=0; i<neighbors.length; i++) {
10. final V child = neighbors[i];
11. if (child.makeParent(this))
12. child.compute(); // recursive call
13. }
14. } // compute
15. } // class V
16. . . . // main program
17. root.parent = root; // Use self-cycle to identify root
18. root.compute();
19. . . .