```
 1: #include "PTree.hpp"
 2:
 3: #define RT sqrt(2)
 4:
 5: PTree::PTree() {}
 6:
 7: void PTree::pTree(RenderWindow &target, int size, Vector2f pos, Vector2f ori
g, int deg, int iter) {
 8:     Vector2f Lp, Rp, origL(0, (size/2)*sqrt(3)), origR(size/2, size/2);
 9:     Color Brown(139, 69, 19);
10:
11:     shape.setPointCount(4);
12:     shape.setPoint(0, Vector2f(0, 0));
13:     shape.setPoint(1, Vector2f(0, size));
14:     shape.setPoint(2, Vector2f(size, size));
15:     shape.setPoint(3, Vector2f(size, 0));
16:
17:     shape.setPosition(pos);
18:     shape.setOrigin(orig);
19:     shape.setRotation(deg);
20:
21:     Rp = shape.getTransform().transformPoint(shape.getPoint(0));
22:     Lp = shape.getTransform().transformPoint(shape.getPoint(3));
23:
24:     shape.setOutlineColor(Color::Green);
25:     shape.setFillColor(Brown);
26:     shape.setOutlineThickness(-5);
27:     if (iter < 0)
28:         return;
29:     Rp = shape.getTransform().transformPoint(shape.getPoint(0));
30:     Lp = shape.getTransform().transformPoint(shape.getPoint(3));
31:
32:     target.draw(shape);
33:     //pTree(target, (size/2)*sqrt(3), Rp, origL, deg-30, iter-1);
34:     //pTree(target, size / 2, Lp, origR, deg+60, iter-1);
35:
36:     pTree(target, (size/2)*sqrt(3), Rp, origL, deg-30, iter-1);
37:     pTree(target, size/2, Lp, origR, deg+60, iter-1);
38: }
```