```
# drawtree.py

from turtle import Turtle, mainloop

def tree(plist, l, a, f):
    """ plist is list of pens
    l is length of branch
    a is half of the angle between 2 branches
    f is factor by which branch is shortened
    from level to level."""
    if l > 5: #
        lst = []
        for p in plist:
            p.forward(l)#沿着当前的方向画画Move the turtle forward by the specified distance, in the direction the turtle is headed.
            q = p.clone()#Create and return a clone of the turtle with same position, heading and turtle properties.
            p.left(a) #Turn turtle left by angle units
            q.right(a)# turn turtle right by angle units, nits are by default degrees, but can be set via the degrees() and radians() functions.
            lst.append(p)#将元素增加到列表的最后
            lst.append(q)
        tree(lst, l*f, a, f)



def main():
    p = Turtle()
    p.color("green")
    p.pensize(5)
    #p.setundobuffer(None)
    p.hideturtle() #Make the turtle invisible. It's a good idea to do this while you're in the middle of doing some complex drawing,
    #because hiding the turtle speeds up the drawing observably.
    #p.speed(10)
   # p.getscreen().tracer(1,0)#Return the TurtleScreen object the turtle is drawing on.
    p.speed(10)
    #TurtleScreen methods can then be called for that object.
    p.left(90)# Turn turtle left by angle units. direction 调整画笔

    p.penup() #Pull the pen up – no drawing when moving.
    p.goto(0,-200)#Move turtle to an absolute position. If the pen is down, draw line. Do not change the turtle's orientation.
    p.pendown()# Pull the pen down – drawing when moving. 这三条语句是一个组合相当于先把笔收起来再移动到指定位置，再把笔放下开始画
    #否则turtle一移动就会自动的把线画出来

    #t = tree([p], 200, 65, 0.6375)
    t = tree([p], 200, 65, 0.6375)

main()
```