1 Administrative Topics

- We take the quiz
- Remember there is a talk today

2 Proj 9

Some advice:

- You can add a line `turtle.hideturtle()` to `Interpreter.__init__` to hide the turtle.

- A strategy for handling color. Set the color in by calling the Interpreter’s `setColor` method. Then, in `drawString`, you don’t need to set the color. But, if you change the color for any reason (e.g. to draw a leaf, you need to set it back. To do this, add two lines to the beginning of the leaf case

  ```python
  clr1 = turtle.pencolor()
  clr2 = turtle.fillcolor()
  ```

  and two lines to the end of the leaf case

  ```python
  turtle.pencolor(clr1)
  turtle.fillcolor(clr2)
  ```
• In your scene code, it is a good idea to add a comment each time you call the draw method on a shape object. This comment should indicate how the x,y position relates to the object (is it the upper left hand corner?). This will make designing and debugging much easier.

• When adding a new capability, e.g. setting the width, it is a good idea to add it to the base Shape class, rather than a child class. The idea is this: a Shape is a flexible, you-can-do-anything shape. And the child classes just make specific statements about what can be done (e.g. a Shape can draw any shape, but a Square can draw only a square).

• Don’t try to do thing that require too many angles.

• Don’t use angles that are too small (I don’t go smaller than 5).

• The test functions for shape and tree should just be functions (not methods). Here is an example of a tree test function:

```python
def test( filename, angle ):
    tree = Tree(filename=filename, angle=angle, iterations=3)
    for x in [-100,0,100]:
        tree.draw( x, 0 )
    terp = interpreter.Interpreter()
    terp.hold()
```