1 3D Turtle

The 3-D turtle uses a 3D-coordinate system in which the positive X-axis extends to the right, the positive Y-axis extends up, and the positive Z-axis extends out of the screen (see Figure 1).

The turtle has its own set of three vectors which determine its orientation. Its forward vector points in the direction it will travel, its right vector points out to its right (like your right arm when fully extended), and its up vector points out through the top of its head. See Figure 2.

The turtle can rotate about any of its three vectors. A rotation around the up vector is called a yaw, which is equivalent to the orientation of a 2D turtle. A positive yaw causes the turtle to rotate to the left (see Figure 3).

A turtle can pitch, which what an airplane does when it is taking off. A positive pitch causes the turtle to rotate around its right arm “backwards” (see Figure 4).

A turtle can roll, which is like tilting. A positive roll causes the turtle to rotate around its forward vector ”to the right” (see Figure 5).

Note that all of these rotations are relative to the turtle’s current position. In other words, a positive pitch causes the turtle to “take off” like an airplane only if it starts in a position with its forward vector parallel to the ground.
Figure 1: Positive portions of $X$-, $Y$-, and $Z$-axes
Figure 2: Stephanie’s picture of a 3D turtle. The black box at the bottom is the pen. The purple “arm” represents the forward vector. The turtle moves in the direction of the purple arm. The orange arm represents the up vector and the light goldenrod arm represents the right vector.
Figure 3: A view from above. A positive yaw (or orientation) is a left turn. I have drawn an arrow to show the left turn as a rotation around the up arm (which is barely visible because we are looking down on it).
Figure 4: A view from the right-hand side. A positive pitch causes the turtle to “fall back”. I have drawn an arrow to show the fall as a rotation around the right arm (which is barely visible because we are looking straight at it).
Figure 5: A view from behind. A positive roll causes the turtle to tilt to the right. I have drawn an arrow to show the right tilt as a rotation around the forward arm (which is barely visible because we are looking at it almost directly from behind).
When the turtle is first created, it is oriented so that its forward vector is aligned with the positive X-axis, its up vector is aligned with the positive Z-axis, and its right arm is aligned with the negative Y-axis. To draw the axes, I create a Line object (which simply draws the string 'F') and add support for setting line widths. Here is the code that draws the axes:

```python
# Draw the axes (y is blue, x is green, z is red)
ln = Line(distance = 300, color=(0,0,1))
ln.setWidth(5)
ln.draw( 0, 0, zpos=0, roll=0, pitch=0, orientation=90)

ln = Line(distance = 300, color=(0,1,0))
ln.setWidth(5)
ln.draw( 0, 0, zpos=0, roll=0, pitch=0, orientation=0)

ln = Line(distance = 300, color=(1,0,0))
ln.setWidth(5)
ln.draw( 0, 0, zpos=0, roll=0, pitch=90, orientation=0)
```

The draw method resets the turtle to its above "initial" orientation, then uses the “orientation” argument to yaw it, then the “roll” argument to roll it, and finally the “pitch” argument to pitch it. I used this code to draw Figure 1.