Progress Report

Nathan Walters
nwalter2

F8: 16 October 2015

I have updated my existing 2D Turtle implementation to represent the turtle in terms of position, heading, and normal vectors. The turtle is performing as expected in my 2D renderer, which simply discards the z-component of the position vector when rendering. I am using `Vector3` objects from `three.js` to represent the turtle's position, heading, and normal vectors.

Using that object and its associated methods has greatly simplified some of the code that controls the turtle's movements. In fact, the `Vector3` object actually contains most the mathematics that will be necessary to implement my project. I have decided to use the built-in mathematical operations in order to write cleaner, more readable code. However, I will still include a discussion of the mathematics in my project documentation.

I can now begin to implement a renderer using `three.js`. To keep the turtle performant, I am going to have to significantly rethink how the renderer works. In its current canvas-based form, the renderer reparses all of the turtle's past commands and recreates the necessary lines for every single frame. However, that could get computationally expensive for complex drawings when using `three.js` since every line requires the creation of a number of objects to describe it. I will have to design a system that avoids the creation of new objects as much as possible. It's never too early for optimizations!