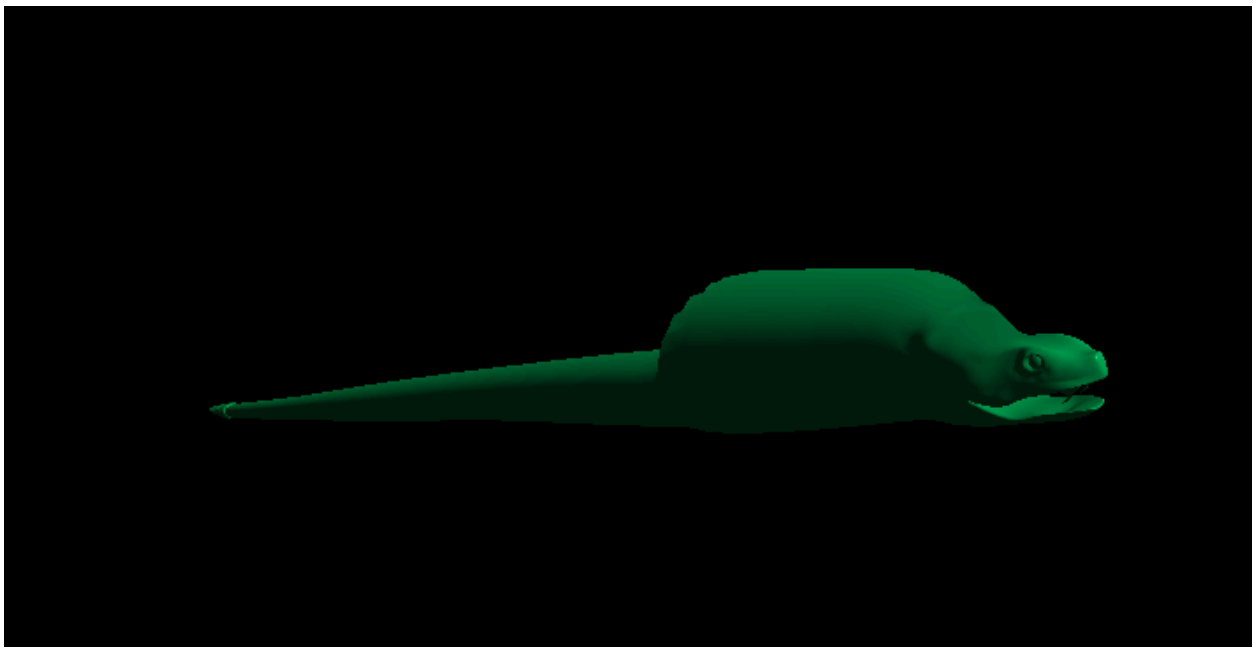
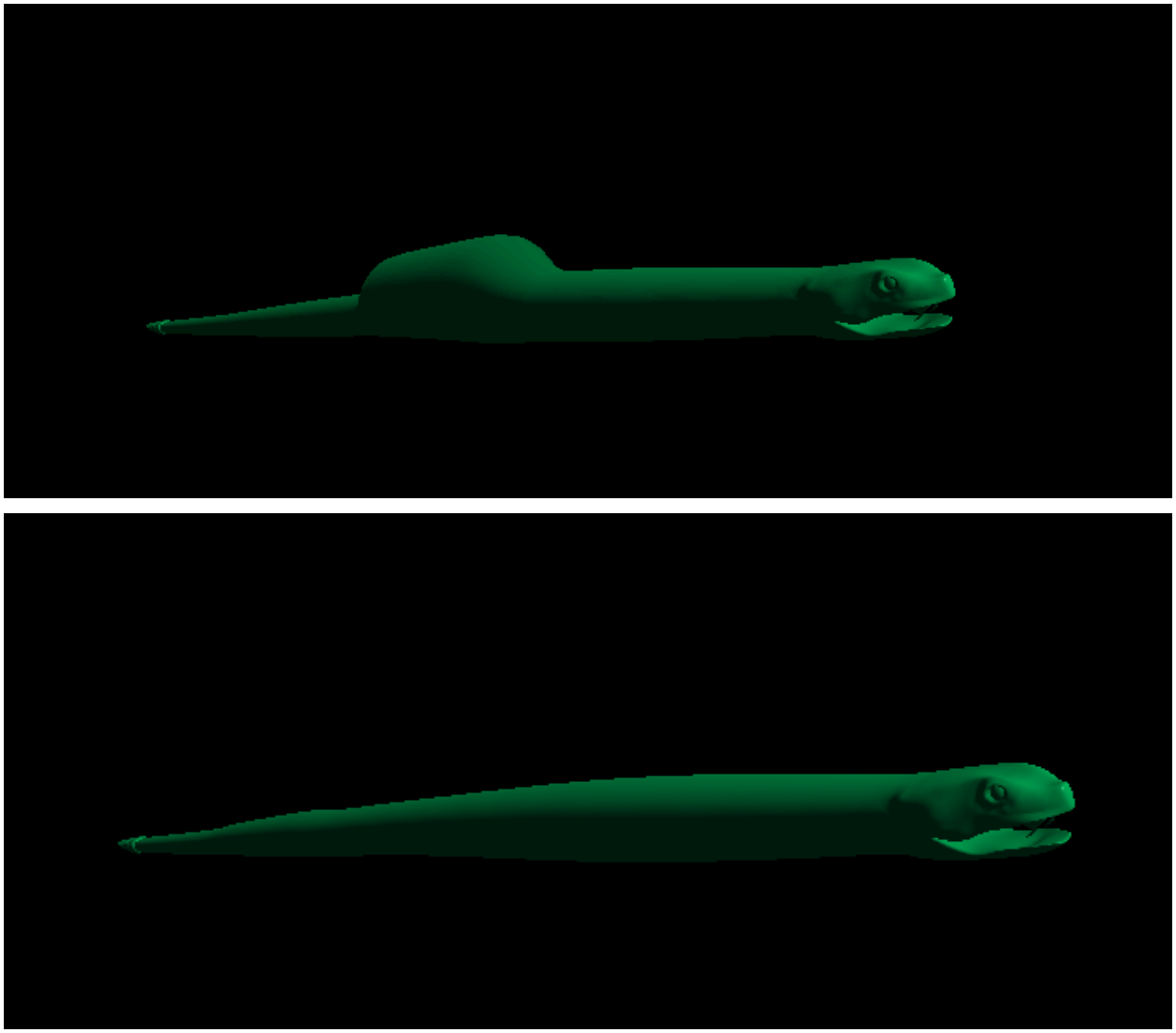
Wolfie Essink
essinkw@oregonstate.edu or wolfie.essink@gmail.com
Project #6 - Shaders, I

What I did:
First off, I started by adding all the necessary files, includes, and global variables that I would need for the project. Next, I created my Snake shader in InitGraphics() by calling Snake.Init() followed by Snake.Create("PigInPython.vert", "PigInPython.frag"). Then, in InitLists(), I loaded the OBJ file using LoadObjMtlFiles() and stored it in a new display list called SnakeDL. This created the actual drawable snake that my vertex shader would later deform for the animation. Now with the object and shader prepared, I moved into the main transformation and rendering logic inside Display(). Here, I computed uPigD, which is the bulge center sliding along the snake from its head to its tail, and uPigH, which is the bulge height, calculated with a sine function to make it pulse smoothly. I then activated the shader with Snake.Use(), sent the two uniforms to the vertex shader using SetUniformVariable(), and finally drew the snake by calling the snake display list. The vertex shader used my uniform values to scale the snake's thickness in its Y-Z plane, creating a smooth traveling bulge.

Screenshots:





What convinces you that your animation is indeed doing what you set it up to do:
I know the animation is behaving exactly as intended because the bulge moves from one end of the snake to the other in sync with the Time variable, and its thickness smoothly increases and decreases according to the sine-based height function. I can also clearly see the vertex shader deforming only the correct X-range of the snake, confirming that the smoothpulse logic and uniform updates are working correctly.

Video:
https://media.oregonstate.edu/media/t/1_d7cnkk7f