CMSC 405: Project #2

Nicholas Mills

CMSC 405: Computer Graphics

University of Maryland University College

Mark Munoz

2/06/2019

**Class Description**

**Concept.** For this project, I chose to make a scene with two major modes. Initially, it is a nice field, with the sun in the distance and a tree off to the side. Underneath the tree sits a box. The user may go investigate the box by walking towards it with the up arrow. When it gets close, a creature pops out of the box and starts to chase the user. The surroundings change to become more menacing, and the creature eventually attacks the user, ending the experience.

**Constants.** Due to the number of modifiable traits of different objects, I created many constants for the sake of clarity and ease of adjustment. These constants include such things as user step size, object coordinates, colors, and sizes, draw distance, window size, and vertical field of view.

**Perspective.** To create a sense of perspective, I used a GLU object to maintain focus on the proper object, either the box or the skull creature depending on what phase the program is in. I made the max view distance very long and the min view distance very close in order to get a good sense of depth. I put borders on the box to make its facings more clear, and I put the sun/eye quite far away in order to make the change of distance as minimal as possible over the movable distance.

**Shape classes.** In order to make the individual objects in the scene more easily manipulable, I created a class structure to support them. All of the shapes are children of the abstract Shape class, which dictates that all shapes have a color and a draw function which draws the basic shape. The major subclasses of this are Rectangle, RectangularPrism, and Circle. Rectangle contains attributes for length and width, which circles contain a radius attribute, as well as logic to draw both of them based on this information. RectangularPrism takes a height and a Rectangle object and draws an appropriate prism based on those dimensions. The two other classes, Box and Skull, extend RectangularPrism, allowing them to make slight modifications upon the draw function for that class, including removing the box lid when the user gets close enough and drawing eyes on the skull creature.

**Test Cases:**

|  |  |  |  |
| --- | --- | --- | --- |
| Purpose | Expected Result | Actual Result | P/F |
| Test initial position | The program will start in a green field, with a blue sky, a green and brown tree off to the left, and a box in the center. All of these objects should be proper rectangular prisms. There is also a circular sun in the distance. |  | P |
| Test Perspective | As the user gets closer, the camera should continue to focus on the box. |  | P |
| Test Mode Shift | When the user gets close enough, the box lid should remove and the skull creature should pop out. The sun should turn into a giant eyeball. The camera should shift to the creature. The color scheme should shift. |  | P |
| Skull Follow | The skull creature should follow the user if the user backs up. |  | P |
| Skull Catch | After a few seconds, the creature should dash at the screen, causing a red splash to appear |  | P |