Sairam Soundararajan

CMSC405: 6381 Computer Graphics (2228)

University of Maryland Global Campus

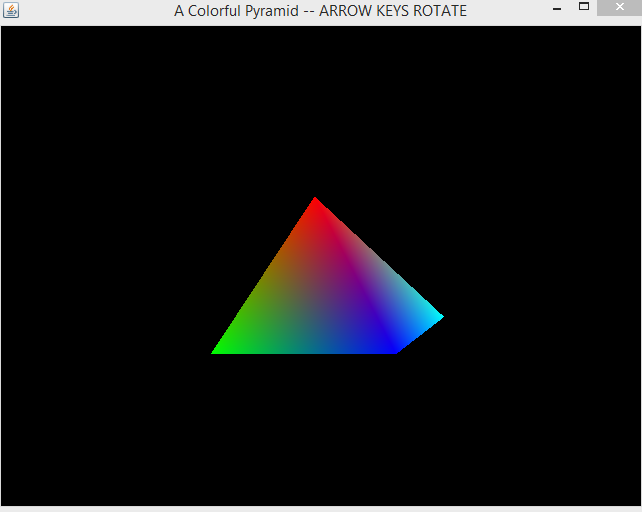
Professor: Lauren King

Date: September 12, 2022

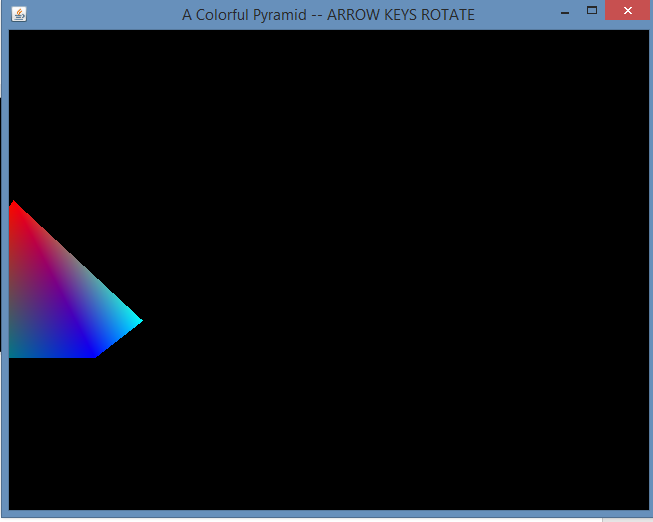
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case | Input | Expected Output | Actual Output | Pass? |
| 1 | Translation | Must translate each shape in x, y, and z direction | Shapes translated in x, y, and z direction | YES |
| 2 | Rotation 1 | Must rotate each shape along the x, y , and z axis | Shapes rotated shape along the x, y , and z axis | Yes |
| 4 | Size | Size of scene must be at  least 640x48 | Size of scene satisfies the 640x48 minimum requirement | Yes |

**Snapshots of Test Runs:**

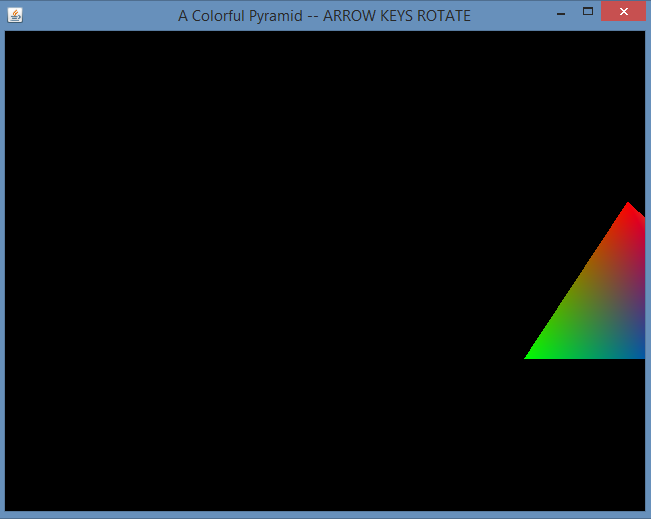
Pyramid at starting point:



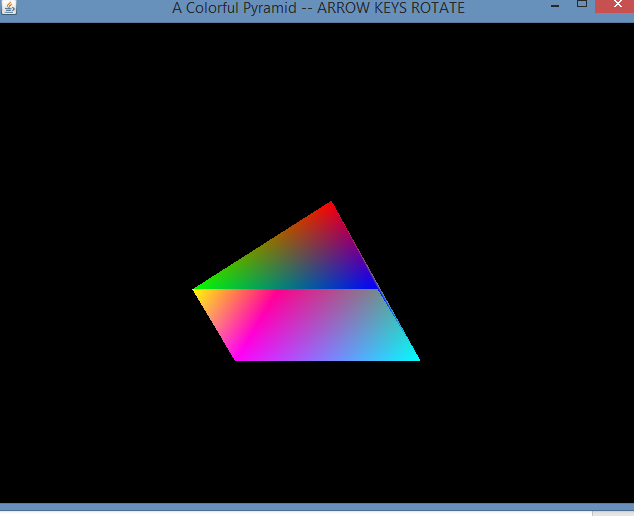
Translated left:



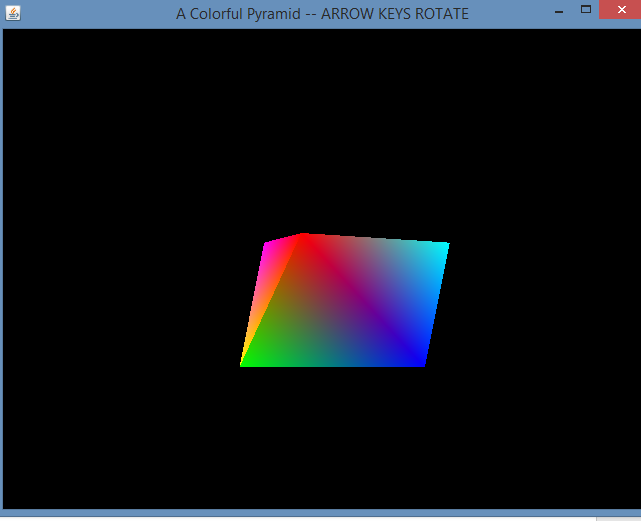
Translated right:



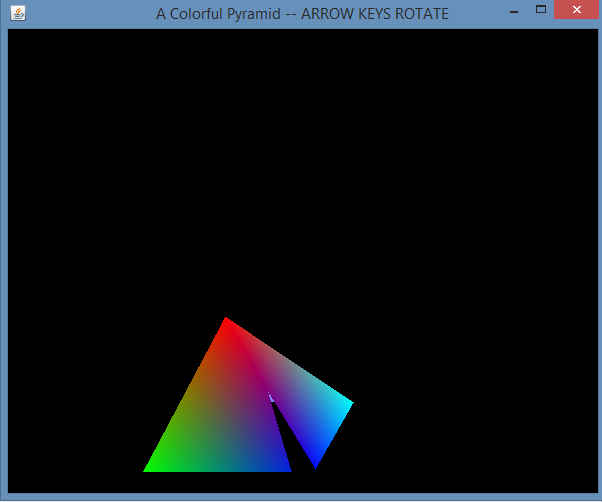
Rotated up:



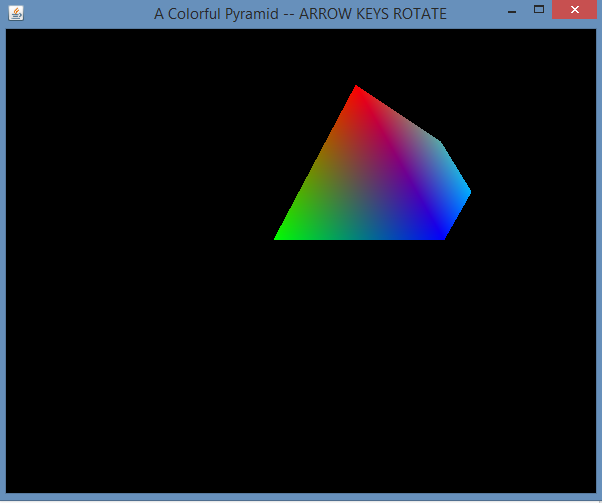
Rotated Down:



Translated left diagonally:



Translated right diagonally:



Lesson Learned:

For this project, I used Eclipse to work on with Java Open GL. I worked on 6 shapes: Cube(from what was already provided), Tetrahedron, Square Pyramid, Rectangular Prism, Hexagonal Prism, and Trapezoidal Prism. In order to form these shapes, I had to rely on entering specific x, y and z coordinates for the necessary amount of vertices for each shape. At first, I thought working on the rectangular prism was going to be simple since it is similar to a cube, but alas, it was actually more difficult because the cube has squares that are consistently same in length and width. Knowing that a rectangle’s length is longer than its width, that was a challenge. I was fortunate to get some tutoring for some of the shapes due to the limited amount of resources on the web. At first, I was thinking about working on a cylinder or a sphere. However, because of the limited resources I had searched for, I ended making a trapezoidal prism instead. Some of the colors, I must forewarn, may fade (depending on where they are in the transformation). But they all work properly with 3 rotations and 3 translations. In a way, I actually like this project because it involves using geometry to code 3d shapes and it was a little more complex than the first project where we used 2D arrays to design 2D shapes. Nevertheless, I would greatly benefit from working on Java GL for a longer period of time so that I can build up to designing even more complex designs in the future.

References

Jogl graphical shapes. Tutorials Point. (n.d.). Retrieved September 13, 2022, from https://www.tutorialspoint.com/jogl/jogl\_graphical\_shapes.htm

Jouvie, J. (n.d.). *Tutorial 04 : Draw simple rotating shapes - render*. OpenGl - Tutorial 04 : Draw simple rotating shapes - Render. Retrieved September 13, 2022, from http://jerome.jouvie.free.fr/opengl-tutorials/Tutorial4.php#Advanced

UMGC. (n.d.). *UnlitCube*. CMSC405. Retrieved from https://learn.umgc.edu/d2l/le/content/719405/viewContent/26836715/View.