CMSC 405: Project #2

Bedemariam Degef

CMSC 405: Computer Graphics

University of Maryland University College

Dr. Amitava Karmaker

2/9/2021

**Test Plan:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case** | **Input:** | **Expected Output** | **Did Test Pass?** |
| 1 | Draw a 3D triangle by calling the drawTriangle() method. It should be located at the top left of the display scene with a few transformations applied. | A 3D triangle rotating in a 90-degree angle about the z-axis at position (-6, 5, 0). | Y |
| 2 | Draw a 3D cube by calling the drawCube() method. It should be located at the top center of the display scene with a few transformations applied. | A 3D cube rotating in a 30-degree angle about the y-axis at position (0, 5, 0). | Y |
| 3 | Draw a 3D cylinder by calling the drawCylinder() method. It should be located at the top right of the display scene with a few transformations applied. | A 3D cylinder rotating in a 45-degree angle about the y-axis, while moving in a continuous circular path. | Y |
| 4 | Draw a 3D plus symbol by calling the drawPlusSymbl() method. It should be located at the middle center of the display scene with a few transformations applied. | A 3D plus symbol rotating in a 45-degree angle about the y-axis, while continuously moving horizontally in a loop from left to right in the middle of the scene. | Y |
| 5 | Draw a 3D tetrahedron by calling the drawTetrahedron() method. It should be located at the bottom left of the display scene with a few transformations applied. | A 3D tetrahedron rotating in a 20-deree angle about the x-axis at position (-6, -5, 0), while also being scaled from very small to its normal size in a continuous loop. | Y |
| 6 | Draw a 3D icosahedron by calling the drawIcosahedron () method. It should be located at the bottom center of the display scene with a few transformations applied. | A 3D icosahedron rotating in a 30-degree angle about the x-axis at position (0, -5, 0). | Y |
| 7 | Draw a 3D diamond at the bottom right of the display scene by calling the drawDiamond() method and applying the necessary transformations. | A 3D diamond rotating in a 35-degree angle about the y-axis at position (6, - 5, 0). | Y |

**Screenshots of successful compilation for Test Cases**

Program is run as a continuous animation with20 milliseconds between the initial and between-event delays.

The orthographic projection view is (-10, 10, -10, 10, -5, 5)

Displays an animation using the seven 3D shapes I created.

Triangle, Cube, Sphere, Plus Symbol, Tetrahedron, Icosahedron, and Diamond.

All the shapes are scaled to fit proportionally in the scene and they are all continuously rotating in different angles. The cylinder in addition is moving in a circular path. The plus symbol in addition is moving in a horizontal line across the window. The tetrahedron in addition is being scaled from very small to its normal size in a continuous loop.

A screenshot of a computer

Description automatically generated with low confidence

Company name

Description automatically generated with medium confidence

A picture containing application

Description automatically generated

A screenshot of a computer

Description automatically generated with low confidence

**Lessons Learned:**

While working on Project 2, there were a lot of new concepts and techniques that I learned about 3D computer graphics, such as how to create and use 3D objects to perform transformations and animations. I had little to no knowledge about using JOGL and OpenGL before this week’s readings thus I had some initial difficulty getting started with the project. I had to thoroughly read through all the reading assignments a few times before I was comfortable enough to understand and complete this project.

For this project, I choose to create 7 different 3D shapes in my scene with each one performing different transformations in the animation. I created the animation using the swing Timer class which helped me create the continuously running effect of the 3D plus symbol being moved in a horizontal line from left to right across the scene. All my shapes are drawn in a separate MyShapes() class and then used in the My3DScene() class to display them.

Overall, this assignment was a good introduction to 3D computer graphics, as it helped me understand and reinforce all the material I learned from the weekly reading assignments. I learned about using openGL in Java and C++ through the JOGL and GLUT libraries. I also learned about creating different types of 3D shapes such as polyhedrons using Indexed Face Sets, and applying colors to them.