

COSC 4370 - Homework 2

Name: Kyle Lumbo PSID: 2170679

March 2025

1 Problem

The assignment requires the creation of several 3D scenes using OpenGL. For the assignment the user must choose three different ways to install OpenGL and be able to compile it, either with Linux (Ubuntu), Windows: (VS2022), or Replit. For my program I have chosen Replit to use OpenGL. The first part of the assignment involves producing three predefined scenes, while the second part involves creating an open-ended scene with your own imagination using OpenGL's transformation mechanisms.

For the second part of the assignment the scene must include at least one instance of nested `glPushMatrix` (a `glPushMatrix` within another `glPushMatrix`) calls and render at least one triangle using OpenGL's immediate mode (`glBegin(GL_TRIANGLES)`).

The goal is to demonstrate proficiency in OpenGL transformations, lighting, and rendering, while also showcasing creativity in the open-ended scene.

2 Method

The implementation involves modifying four parts, each implemented in a separate function:

1. Problem 1: Create a rhombus pattern using teapots facing inwards to the center.
2. Problem 2: Construct a pyramid using stacked cubes.
3. Problem 3: Build a pyramid of teapots.
4. Problem 4: Create an open-ended scene--- in my case I have created an Iron Man Repulsor Hand Cannon.

Each problem uses OpenGL's transformation functions (`glTranslatef`, `glRotatef`, `glScalef`, etc.) to position and shape objects in a 3D space. The open-ended scene (problem 4) also includes nested transformations and immediate mode rendering of a triangle to represent the hand cannon.

3 Implementation

Problem 1: Rhombus pattern with Teapots facing Inwards to the Center

The problem1() function arranges eight teapots in a rhombus pattern. Each teapot is rotated and translated to face the center of the rhombus.

The positions of the teapots are calculated using a 2D array, and each teapot is rotated by 45 degrees around the Y-axis to create the rhombus pattern. The key steps are:

1. Define the positions of the eight teapots in a 2D array.
2. Use glPushMatrix and glPopMatrix to apply transformations for each teapot.
3. Rotate each teapot by 45 degrees around the Y-axis to create the rhombus pattern.
4. Translate and rotate the teapots to face inward toward the center.

Problem 2: Pyramid of Cubes

The problem2() function constructs a pyramid using stacked cubes. Each level of the pyramid is scaled and translated to create the pyramid shape. Each cube is stretched to create the next level of the pyramid. The pyramid is constructed by stacking cubes, with each level scaled and translated to create the pyramid shape. The key steps are:

1. Uses a loop to create a specific number of levels of the pyramid. (in this case 25 levels)
2. Translate each level upward to create the pyramid effect.
3. Scale each cube to create each level of the pyramid.
4. Use glutSolidCube to render each level.

Problem 3: Triangle of Teapots

The problem3() function builds a triangle using teapots. Each row of the triangle contains an increasing number of teapots. The teapot triangle is built using nested loops to position teapots in an upside-down triangle shape. The key steps are:

1. Use nested loops to position teapots in a triangle shape.
2. Translate each teapot to its correct position in the triangle.
3. Using yOffset to make the triangle upside down.

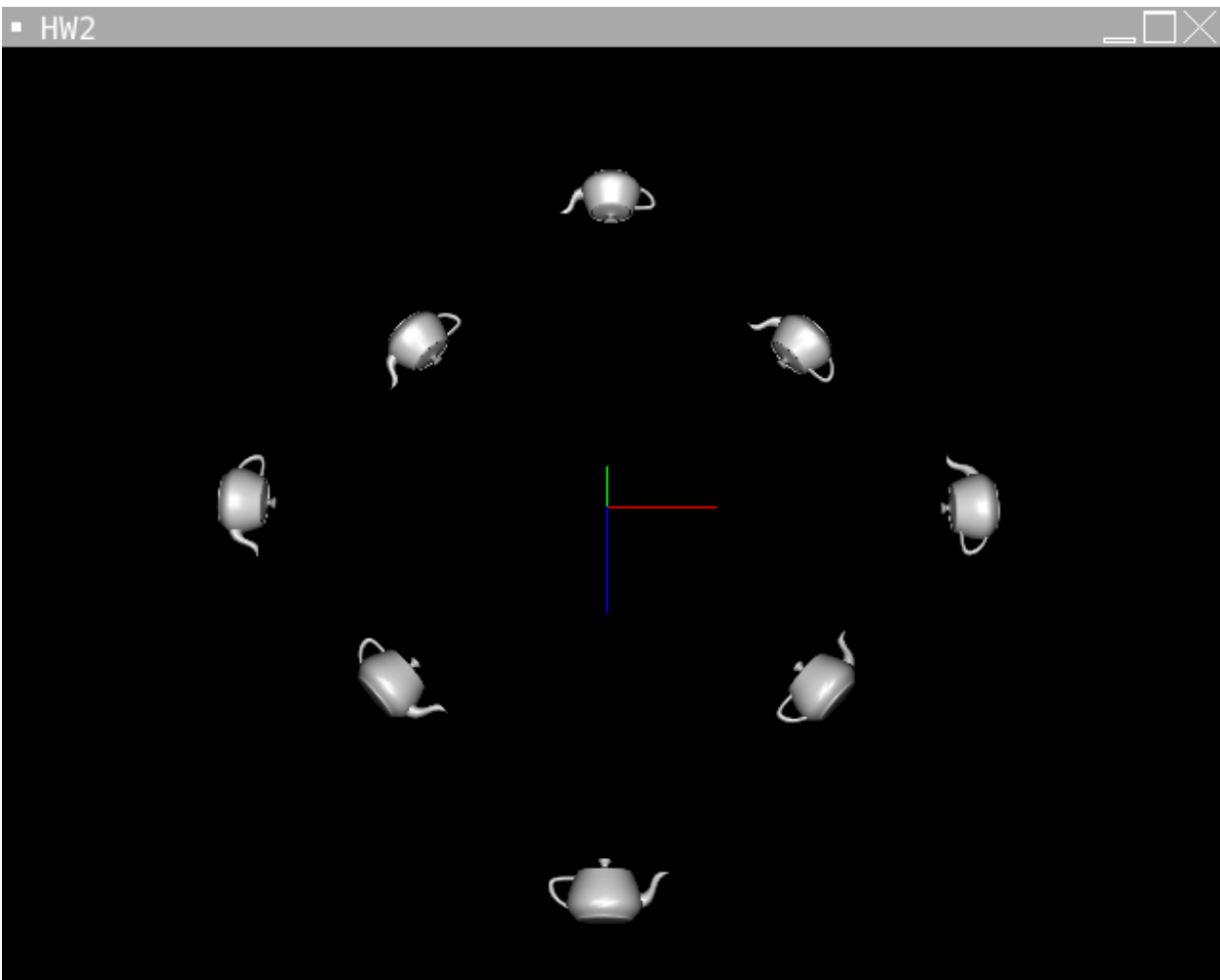
Problem 4: Iron Man Repulsor Hand Cannon

The problem4() function creates an open-ended 3D object. In my case, I have created a 3D model of an Iron Man Repulsor Hand Cannon. The scene includes a palm, fingers, a thumb, and a Repulsor Hand Cannon represented by a triangle. The Iron Man Repulsor Hand Cannon is created using nested transformations for the fingers and thumb, and a triangle is rendered using immediate mode to represent the hand cannon. The key steps are:

1. Use `glPushMatrix` and `glPopMatrix` to apply nested transformations for the fingers and the thumb.
2. Render the palm and fingers using `glutSolidCube`.
3. Use `glBegin(GL_TRIANGLES)` to render the hand cannon as a triangle.

4 Results

The output of the program is a series of 3D scenes rendered in an OpenGL window. The user can switch between the scenes using keyboard inputs 1,2,3, and 4. The open-ended scene (Problem 4) successfully demonstrates nested transformations and immediate mode rendering of a triangle, fulfilling the assignment requirements.



HW2

