

COSC 4370 – Homework 2

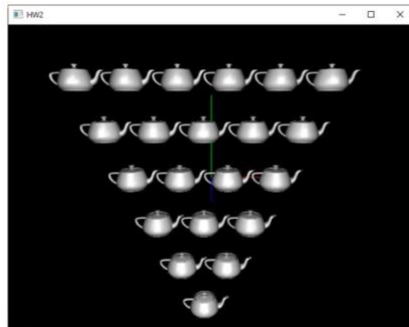
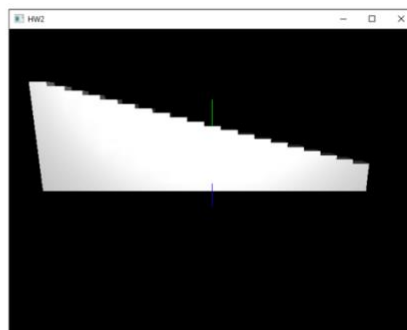
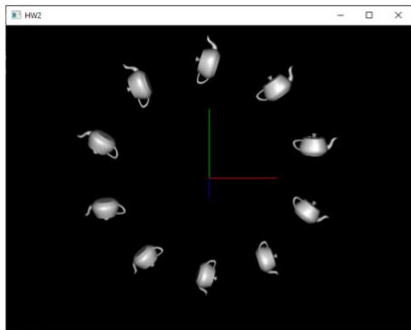
Daniel Nguyen 1479087

October 2022

1 Problem

1.1 Problem

Recreate the three images given using OpenGL.



1.2 Problem

Create a scene of your own imagination using similar OpenGL technique.

2 Method

2.1 Method

The techniques that were used in the problems are `glPushMatrix`, `glPopMatrix`, `glTranslatef`, `glRotate`, `glutSolidTeapot`, `glutSolidCube`, and `math`. `glTranslatef` is used to move the our current coordinate on the x,y,z axis. `glPushMatrix` allows us to save the current coordinate when this is called. `glPushMatrix` is used together with `glPopMatrix` which loads the save point or load the last `glPushMatrix`. For example, let's say the current coordinates are (0,0,0) when

we call `glPushMatrix` it saves the (0,0,0) coordinates. Then we `glTranslatef(-1,1,2)` which moves the coordinates to the new coordinate (-1,1,2) when we are done with this coordinate we can `glPopMatrix` which would load the last saved `glPushMatrix` which is (0,0,0). `glRotate` is used to rotate the object along the x,y,z axis. `glutSolidTeapot` and `glutSolidCube` is used to create an object on the current coordinate teapot being a shape of a teapot and cube being the shape of a cube. Math to calculate the position and how much to `glTranslatef`.

3 Implementation

3.1 Implementation

First image

Looking at the given image the main object will be `glutTeapot` that is rotated along the z axis circling around the coordinate (0,0,0). There are 10 teapots in the scene so to evenly distribute the teapot in a circle I divided 360 by 11. Why 11? Because if I divided 360 by 10 then there will be an overlap when the coordinate is (1,0,0) given if the radius is 1, displaying only 9 teapots. Using the equations $x = \text{radius} * \cos(\text{angle})$ and $y = \text{radius} * \sin(\text{angle})$ to find the position of the teapots in. We find the angle by starting at 0, then adding 360/11 after every iteration of a for loop. We found each position of the teapot, so we use `glPushMatrix` to save our current position then `glTranslatef(radius*cos(angle), radius*sin(angle), 0.0)` to the location the teapot would be and call `glutSolidTeapot`. The last step would be to `glRotate(angle, 0.0, 0.0, 1)` the pot by the z axis and call `glPopMatrix` defaulting to (0,0,0).

Second image

Looking at the given image the main object will be `glutSolidCube` stacked on top of each other creating stairs. There are a total of 20 steps and the steps looks like they half of half of the cube. In other words, the steps are $(\text{the size of the cube}/2)/2$ in this case $(.1/2)/2$. This will give us the y difference of each step. Using a for loop nest I can layout the x axis and lay the y on top increasing the number of cubes stacked on top as the x decreases. The first cube will always start with $y=0$ and will iterate up by 0.025 per every cube added, this will mean that we would need to reset the y to 0 after every for loop. I used `glPushMatrix` to save the default (0,0,0), `glTranslatef(##,##)` to the given location depending on the iteration, then `glPopMatrix` to restore default (0,0,0) and repeat.

Third image

This image is forms an upside-down triangle using `glutTeapot`, there are 6 layers with each layer incrementing up by 1. Looking at this scene we can see that every odd will have a teapot that $x=0$. With that we know that every 2 iteration we there must be a change in x where x equals 0. One solution is that we can take the change of x and then multiply it by 2 and that will give us the number we should be adding to the change of x. We add in this case because x is negatively changing every layer. Using `glPushMatrix` we can save the default position (0,0,0) then `glTranslatef()` using the equation and the `glPopMatrix` back to the original position.

3.2 Implementation

For the fourth problem I tried to create a face, using cubes for everything except the nose. I manually inputted the coordinates to create 3 triangles creating a 3d nose that would stick out on the -z axis. Like the previous problems I used `glPushMatrix` and `glPopMatrix` to save and load the coordinates (0,0,0) so I know what position I am working with. `glTranslatef` to evenly place the eyes and the circle head using the previously mentioned equation $\text{float } x = (\text{radius} * \cos(\text{angle}))$ and $\text{float } y = (\text{radius} * \sin(\text{angle}))$. For me to be able to place triangles I used `glBegin(GL_TRIANGLES)` indicating that I will be feeding points to create triangles once the nose has been created I close the `glBegin` with `glEnd()` indicating I finished feeding the points.

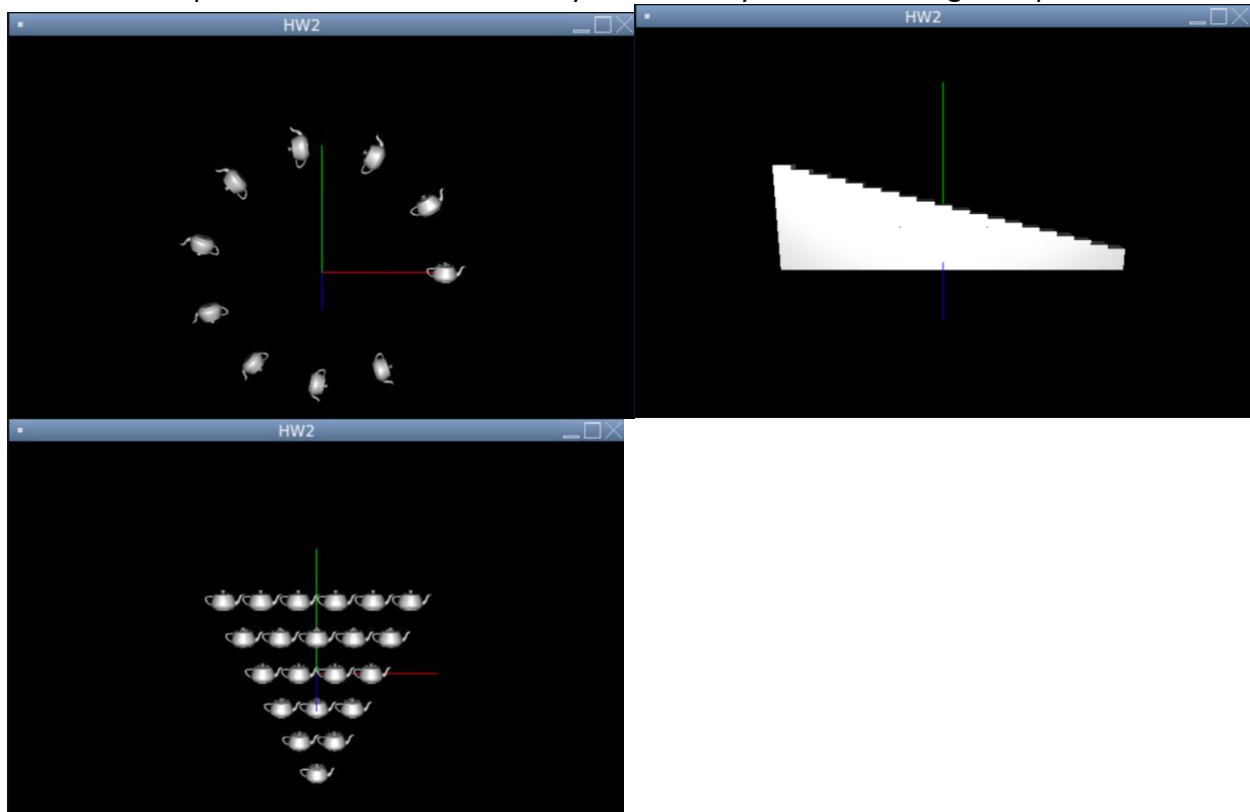
4 Results

4.1 Result

The output of problem 1.1 shows teapots in a circle with angles changed on the z axis depending on where the position of the teapot is in the circle.

The output for problem 1.2 Shows 20 steps of stairs descending from -x axis to x axis there are more cubes stacked on top of each other on the higher steps and has a minimum of two steps on the lower end

The output for problem 1.3 shows an upside down triangle of teapots with every layer the number of teapot on the row is increased by 1 with every odd row having 1 teapot with 0 x axis.



4.2 Result

The output of my creativity is a face made of mainly cubes creating the circle face, the eyes, and the mouth. The nose is made of 3 triangles making the nose more 3D like.

