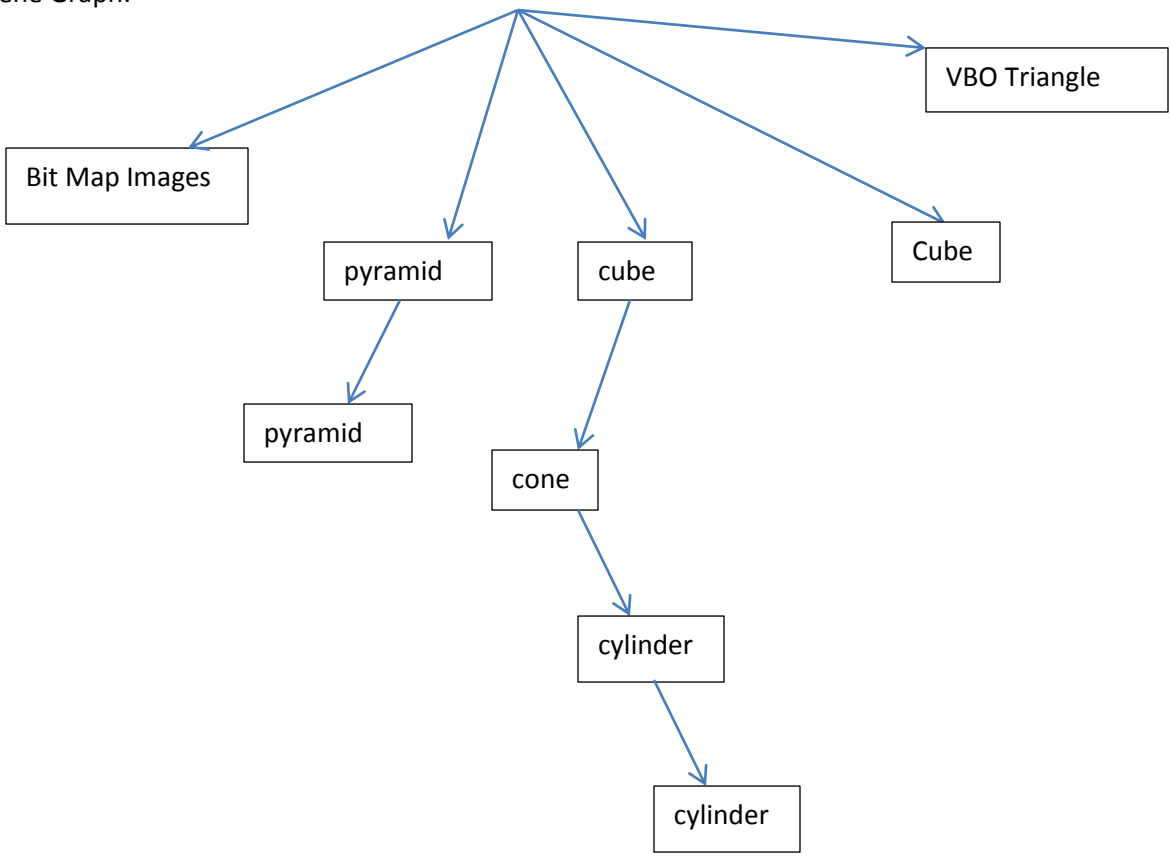
PROJECT B EECS 351

Goal:

The goal is create a program that uses a tree of transformations to render a scene. Then four cameras are placed in this scene and it is drawn on four viewports. The four viewports show a front view, side view, top view and a perspective projection view. There are four image bitmaps that are rendered and movable by arrow keys.

Scene Graph:



The Four Viewports are:

- Front View (top right)
- Side View (top left)
- Top View (bottom left)
- Perspective View (bottom right)

Control Panels:

- ESC – exit program
- Camera Controls:
    - W – forward
    - S – backward
    - A – Turn Left
    - D – Turn Right

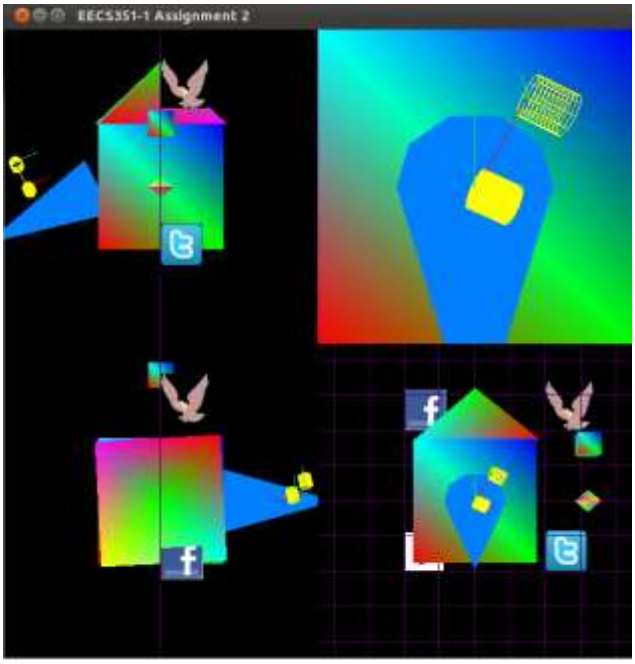
    - X – Turn up
    - Y – Turn down
    - V – strafe right
    - C – strafe left
    - R – move up
    - F – move down
    - M/N – roll
- L – toggles between bottom right projection from perspective and orthographic
- Bitmap image controls:
    - U – change selection of image
    - Arrow keys used to move that selected images
- Asymmetric camera controls(increase/decrease):
    - g/e – change left
    - j/k – change right
    - t/z – change bottom
    - o/q – change top
- p – pause animation
- h – print help

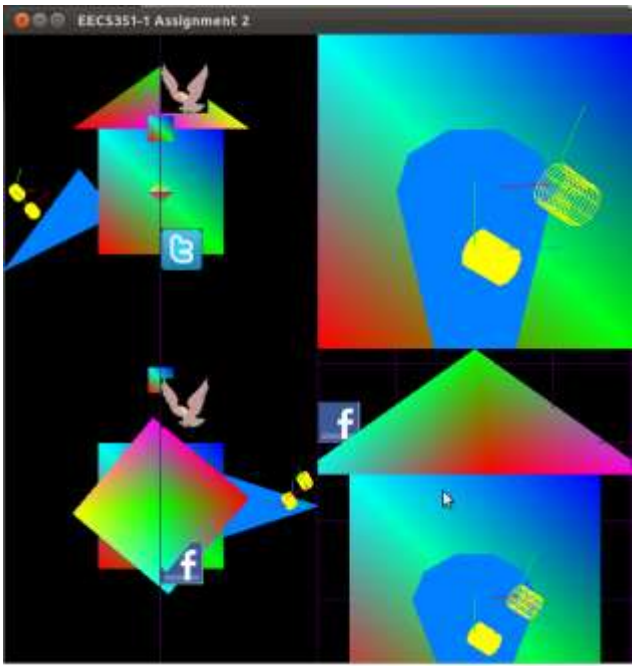
Code Guide:
Main.cpp contains all the definitions of the main OpenGL functions. Cube.h has all the properties of the program. Shapes.h contain class definitions of all the 3-D shapes.

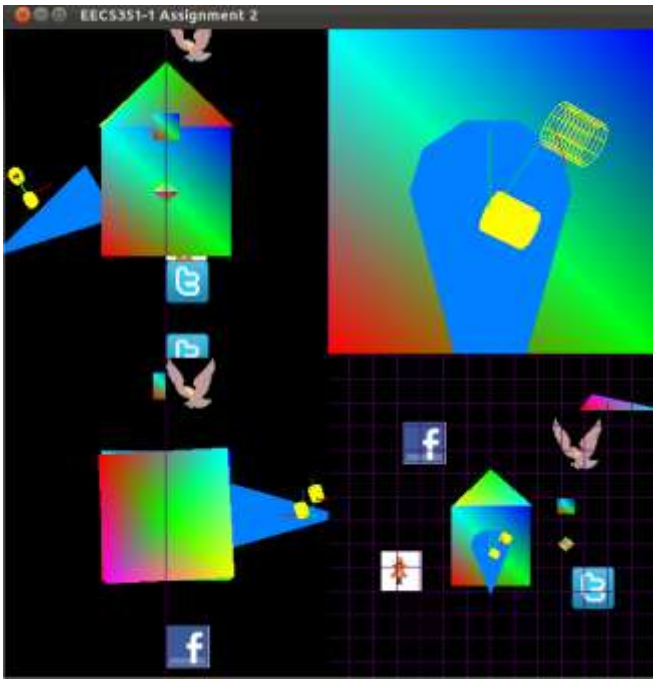
Updates from the previous version shown on demo day:

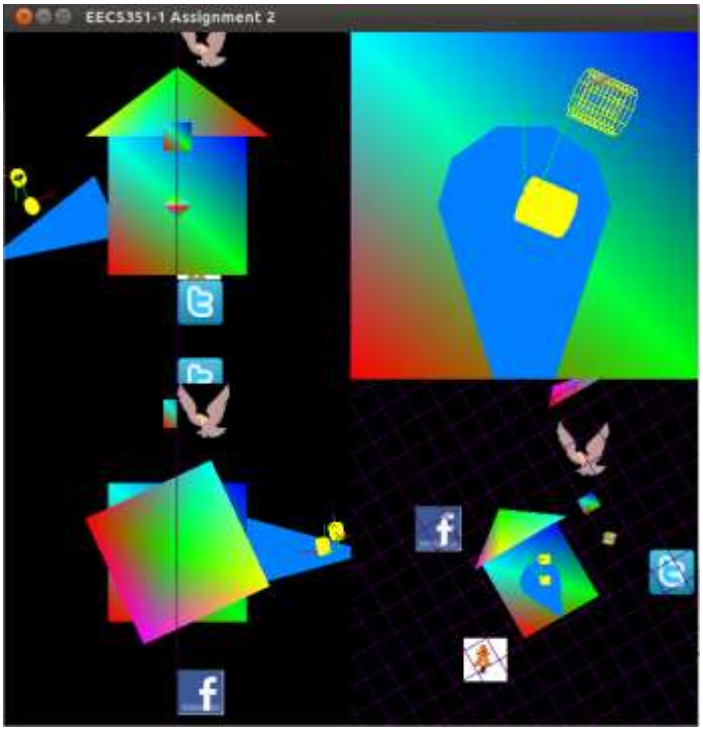
- Asymmetric camera adjusts have been added. Press the corresponding controls
- Vertex Buffer Objects have been done for one shape namely CTriangle found in shapes.h.

Screenshots:






EECS351-1 Assignment 2


EECS351-1 Assignment 2