**Programming Project Report**

Rhett Brandon

010722727

**Problem Statement:**

The goal of this project was to create a program that made it look like it was “raining cats and dogs”. This project asked for no inputs, but the output should create random dog or cat image blocks and have them fall from the top of the screen, simulating rain. No error handling was required. The program must be ran with the libim directory, please use the comp\_proj.sh command to compile the project it will need to be ran like:

***./comp\_proj.sh project\_5***

This will create a project\_5 executable from the project\_5.cpp file.

**Design:**

Upon designing this program I had to figure out the best way to store the 20 images of cats and dogs. I decided to load all of the texture arrays from the images and put them into their own array, so an array of texture arrays. This was ultimately decided to save on space in the creation of the block object array. The next big design decision I had to make was to create a way to store information for each individual block. In order to do this I created a block object that would randomly instantiate different values for the block position, as well as the image displayed on the cube, and the rate at which the cube should fall. Instead of storing each texture array in the object or setting a pointer from the block object to the texture array, I decided to have the block object store a number between 0 and 19. The randomly assigned number would then choose which texture array to load. Using a object oriented approach was useful since it helped compartmentalize the program when troubleshooting, the downside is that it complicated things when I added more objects to the program. Upon adding more object to show on screen I noticed that there were some odd graphical things going on where blocks would disappear when they reach the middle of the screen. Lowering the number of objects actually caused less of a graphical but still showed enough objects to get the point across.

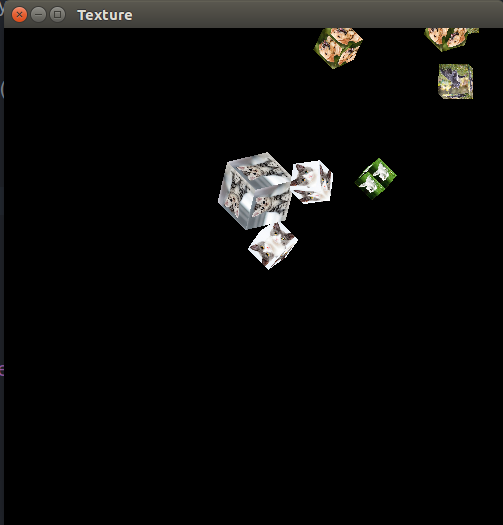
**Implementation:**

To begin the implementation of this project I started with the texture3.cpp program, this program initially wrapped a brick image around a 3-d cube. From there I started by attempting to load in the cat-dog textures to try and display them on the cube. Once the object was displaying I decided to begin implementing the randomness the program called for. I wanted to test and make sure that every time I reran the program a different picture was shown. Then I began animating the single cube by starting with the simulated size of the cube, and the gravity. Once gravity was working I started to work on the cube rotating. Upon all of this working I began implementing multiple cube objects being displayed at once.

* This section should be 1-2 paragraphs long.

**Testing:**

In order to test the program I incrementally tested as I implemented each feature. For instance, I started by ensuring that I could display a cat-dog picture onto the cube. Then I tested to see if I could display a random image each time I ran the program. Next I validated that the animations were working on the cube, and finally made sure that the program worked with multiple cubes. I am not super satisfied with how the multiple cubes are displaying and animating, but I think the purposes of this project it does what it needs to.



**Conclusions:**

The overall result of this project was a success. The project simulated dog and cat blocks falling from the screen, although sometimes it rains sideways. The next time I implement this I think I would spend more time on the random number generation, I think this contributed to most of my issues when troubleshooting. Overall, the project only took about 8 hours.