Programming Project Report #5

Pablo Guerra
010849976

Problem Statement:

For our fifth project we were tasked with the job of making it rain cats and dogs. That is, cubes falling downwards with images of cats and dogs on them. The project was broken down into several tasks which included:

- Reading Texture Images
- Model Creation
- Model Display
- Animating the model

Design:

In order to give the impression that it is raining cats and dogs, we must take into consideration three main factors. The cubes should fall at different speeds, different sizes and should have both cat and dog images. To achieve this, an array of cube structs were created. Each cube had randomly generated parameters which include; position, velocity, rotation, radius, and image texture. With these random values, we can ensure that each cube generated will look and move differently. There were two extra functionalities implemented in the program to allow for easier testing and viewing. The fist function added was the ability to pause and play the motion of the cubes by pressing the spacebar and the second feature allows for cubes to be regenerated at the top of screen after falling below a certain threshold.

Implementation:

The development of the project began by first running and testing the provided texture3.cpp example. This program contains a cube with a brick wall texture that can be rotated and translated using keyboard inputs. This example program was used as a foundation for the rest of the project. Firstly, the program was modified to read and display images of a cat on the cube instead of the brick wall. Next, a cube structure was implemented which simplifies the process of creating multiple cubes. These cubes were added into an array and displayed all at once on the screen. Lastly, the motion, size and image of the cubes were implemented by generating random variables for all the cube parameters.

Testing:

The program was tested after completing each individual step to ensure that any bugs introduced in that step were solved immediately. This allowed for easier debugging and changes to the newly implemented functions. Since there isn't much inputs for the program, it was easy to test all the cases and see what needed to be fixed. Each step of the project was not started until all the bugs were fixed for the previous steps. Below is a sample image of the final product.

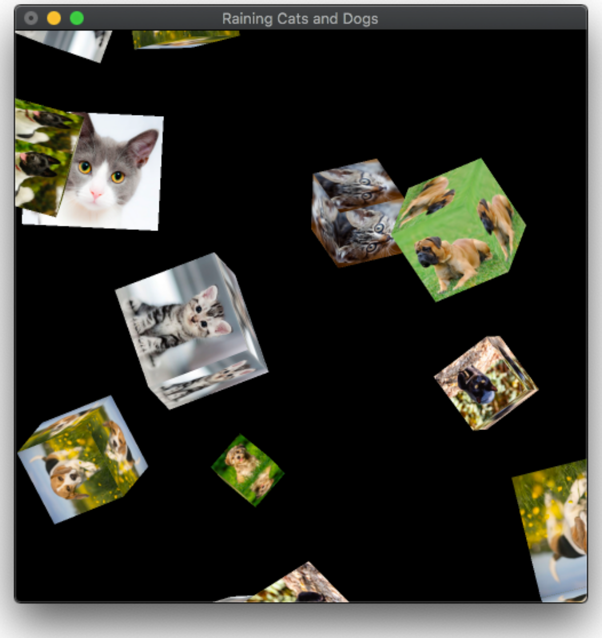

Image showing sample of program output

Conclusions:

Overall, the program works as intended and the different sized cubes with random images fall down the screen while rotating. The creation of new cubes at the top allow for easier viewing since they fall continuously. The one thing I would do differently is to have the cubes rotate in their center point as supposed to a corner. This would provide for a much better-looking animation. The provided example programs are also a great help when working on project like these since it takes most of the guess work and trial and error out of the process.