willemsn

OpenGL - Lab 1

OpenGL - Getting Started

LEARNING OBJECTIVES

- Understand the basic terms used to describe graphics hardware.
- Execute a basic OpenGL application on your lab computer.
- Modify an OpenGL program to change the screen color over time.

This lab will provide you an initial introduction to using modern OpenGL. The follow-up labs in the coming weeks will extend your knowledge about programming the graphics hardware pipeline with OpenGL.

First, read Sections 16.1 - 16.3. These sections discuss graphics hardware in general. Then answer the following questions.

1. What is a term used to describe the parallel computing cores found on modern graphics hardware?
2. What does the term host mean as related to graphics hardware?
3. What does the term device mean as related to graphics hardware?
4. In a graphics hardware system, what is the graphics context?
5. Why is programming OpenGL considered heterogeneous multiprocessing?

Next, fork the following repository from the University of Minnesota github system:

`https://github.umn.edu/willemsn/OpenGL_Lab1`

Activity: If you already know a little about git and the University of Minnesota's gituhub site, pair up with someone in lab who doesn't. If you are unfamiliar with git and/or the University of Minnesota github site, pair up with someone who is familiar.

Help each other fork their own copies of this repository. This is so that each of you can make modifications to it and store those changes back up on the UMN's github site. Next, clone your newly forked repository on to your lab machine. Again, work with someone and make sure they explain what they know to you about using git and github.

Answer one of the following depending on your answer to the "if" in the question:

6a. If you already knew about the UMN github and git, who did you help and what did you teach them about git?

6b. If you didn't know much about git or have never used the UMN github, who helped you and what did they teach you about git?

Now that, you have the code, let's build it. To build this starter code, you will need to use cmake. Here are some instructions to get you going:

First, open up the terminal and change the directory to the name of your repository (called MyOpenGLLab in the following example - use the name you gave it):

```
cd MyOpenGLLab
```

Next, create the cmake build directory to keep your source separate from your build:

```
mkdir build
```

Change into the build directory and then run cmake:

```
cd build
cmake ..
```

This should cause cmake to create the Makefiles for your project. Build it.

```
make
```

You can then run the example by typing:

```
./OpenGL/glfwExample
```

You should see a blank screen that is sort of greyish in color. The program quits when you press the Escape key.

Read Sections 16.4 through 16.6 and answer the following questions:

7. How is the notion of the framebuffer from your first assignment applied or used in OpenGL? After you come up with your own answer, write it down and the confer with your neighbors to see if you all agree.

8. How do you clear out the OpenGL display buffer to zero? What color is this "zeroing out"? How would you clear the display buffer to all red?

9. How many times does the OpenGL code I provided change the state of the glClearColor?

10. Together with a partner, modify the code I provided to cycle through different background colors live while the program runs. The changing background color should happen automatically. *Hint: what state do you need to change and where do you need to change it? And finally, how do you need to change it so you can see it happen?* Here's a personal goal for you to try - can you cycle through all of the RGB color space? Once both you and your partner have completed this in each of your code bases, commit and push your changes back to your repository on the UMN's github and show the TA your changing screen to get all the

points.