

CS4815 Week02 Lab Exercise

Lab Objective: The object of this week’s lab is to take a first look at **OpenGL**, our graphics library and to compile and run a program. Here’s a quick summary of the tasks:

- ❶ Create a series of hierarchical directories to manage our work throughout the semester
- ❷ Copy this week’s lab from the class directory `~cs4815/labs/week02`
- ❸ Examine the `gasket.c` file there using a file browser or **emacs** or whatever editor you like to use
- ❹ Compile and run the program and play around with the number of iterations that it runs for
- ❺ Modify the colour that the program draws with
- ❻ Submit your lab for marking using **handin**

In Detail

❶ Note that there are a few alternatives given below for this first step so please read the entire instructions before acting. Over the semester we will have labs, programming assignments, etc. and it is a good idea to keep these in an organised way. My suggestion is that you create a subdirectory of your home directory called **cs4815** that will be the top-level point for all module-related material. The command to create a directory in Linux is **mkdir**, so you could do

```
mkdir ~/cs4815
```

which makes a subdirectory located in your home directory. Next you should make a subdirectory called **labs** in this for each week’s work. This can be done with

```
mkdir ~/cs4815/labs
```

Finally, for this week’s lab, week02, you should create its own subdirectory with

```
mkdir ~/cs4815/labs/week02
```

An alternative to making each level of the hierarchy at a time is to tell **mkdir** to make the “parent” subdirectory if it doesn’t exist. So the following command can take the place of all of the previous ones.

```
mkdir -p ~/cs4815/labs/week02
```

The `-p` is for “make parents if they don’t already exist.” Note that it is very like the command before it, but you had to do a lot of previous work in order to achieve that.

Yet another alternative is to bring up a file manager and use the “Create ...” menu option there.

② With the semester’s hierarchy set up, now copy to your `~/cs4815/labs/week02` directory the program we will be examining today. This is called `gasket.c` and it can be found in `~/cs4815/labs/week02` sub-directory. Change your working directory into this week’s lab directory with

```
cd ~/cs4815/labs/week02
```

The command to perform the copy is

```
cp ~/cs4815/labs/week02/gasket.c .
```

Note the dot at the end of the command that signifies copying it into the current working directory and keeping the name the same.

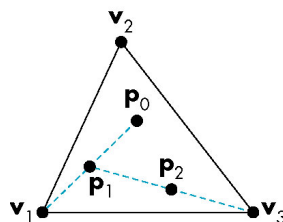
③ The program you are looking at is, by graphics programming standards, a pretty straightforward one. Don’t worry if it is confusing to you now, the object of the lab is to give you a chance to see one from the inside.

The program generates a very famous graphic called the *Sierpinski Gasket*. Later you can compile and run the program and see the resulting output but first we will briefly discuss what the program is doing.

Given a triangle comprising three points in the xy -plane the construction goes as follows:

1. Pick an initial point at random inside the triangle select one of the three corners at random
2. Find the point halfway between the initial point and the randomly selected vertex
3. Display this new point by putting some sort of marker, such as a small circle, at its location
4. Replace the initial point with this new point
5. Return to Step 2

The first two iterations of the algorithm are shown below; in the program the loop repeats 5000 times, as shown in `for` loop of line 43, in the function `display()`.



It would be reasonable to think that this algorithm would just create a random smattering of points inside the given triangle. We will compile the program now and see what happens when we run it.

④ In order to compile the program you will need to open a “terminal session” so that you can issue commands. Follow one of the menus in the top-lefthand corner and you should be able to open a terminal, if there isn’t one opened by default.

The command for compiling the program is

```
gcc gasket.c -l GL -l GLU -l glut -o gasket
```

Note that that’s not a “dash 1”, it’s “dash l” (short for library file).

Linux is always sensitive to character case so be careful to type it exactly as shown. The program should compile without errors. After that you can run it with the command

```
./gasket
```

Don’t worry about the error reported to the screen (if any). Play around with the number of iterations that the program runs for and see what effect it has.

⑤ The final part of the lab exercise is to change the colour that it draws the points in. The default drawing colour, red, is given by the program statement

```
glColor3f(1.0, 0.0, 0.0);
```

on line 14 in the function `myinit()`. Click [here](#) and follow the menu items to `glColor` for documentation for an explanation of `glColor3f(1.0, 0.0, 0.0)`;

You should now modify the colour that it draws in to your own “private” colour using the *last* 6 digits of your student ID. If your student ID is 11051234 then for the RGB values you should use `051234` (and poor old “11” at the front gets ignored). Use these three integers in the range of 0 to 99 and by dividing each by 100 you will get a value between 0 and 1 as required for each of the basic colour components.

Make this change to the function call `glColor3f(1.0, 0.0, 0.0)` and recompile. Please leave the executable and `.c` file in your directory so that you can hand it in for inspection using the `handin` program. And speaking of `handin`...

⑥ Your lab work, like all CS4815 labs this semester, will be due no later than 09.00, Thursday next. This will give everybody as close as is practicable to a week to work on each one. This will be the continuing pattern throughout the semester, too¹: the lab sheet that you first work on in Week n will be due no later than 09.00, Thursday, Week m , $m = n + 1$.

Below I will give you the command that you will need to run in order to submit your work. In the early weeks you will get marks for simply handing something in and having it compile. Your lab work will also be visually inspected by me later to make sure it does what

¹with one or two exceptions

it is supposed to do. Initially this won't count for much but as the semester goes on, this is where most of the marks will be allocated.

The command you need to run in order to make your submission:

```
handin -m cs4815 -p w02
```