

## EBU5405: 3D Graphics Programming Tools – Lab 3

### 1. Introduction

In this lab you will start working in 3D.

### 2. Drawing 3D shapes

You should download and open the Week2Code&Exercises.doc file.

Copy and paste each sample code in its own .c file and name each .c file as shown in the Week2Code&Exercises.doc file.

Compile and run each program.

Now try answering the questions or exercises below.

a) colourCube.c

- i. Look for all the statements in the code that are responsible for double buffering (smooth animation) and hidden surface removal (z-buffer algorithm). Remove these statements (and replace them by others if necessary) and observe what happens.
- ii. Remove the statement that requests the flat-shading colour model. What happens and why?
- iii. Move the *glColor3fv* statements in the *colorcube()* function to another function so you can observe some smooth shading effects on the cube.

b) barnWithWindows.c: do Exercise 1

c) SierpinskiGasket2D.c

- i. Change the number of recursive steps to 1, then 2, then 8, then 20. What happens and why?
- ii. What is the maximum number of recursive steps that your system seems to be able to cope with?

d) SierpinskiGasket3D.c: Modify the code to get the 3D Sierpinski Gasket to spin in the same way the colour cube does (you should use code from the colourCube.c program).

e) sphere.c:

- i. do Exercise 2
- ii. Increase the value of n (number of subdivision steps) until the sphere appears to be solid.

f) twist.c

- i. do Exercise 3
- ii. Try the following values for the twist : 0.5, 1.5, 3.0, and 5.0.
- iii. Now try different values of the twist for the following values of n (number of recursive steps) : 2, 4, and 8.

g) wireframe.c

- i. Refer to the GLUTObjects.pdf file (on Blackboard) and understand the arguments of each of the GLUT object declarations.
- ii. Modify some of these arguments to change the size, radius, height, etc. of some of the objects.
- iii. Comment out the gluLookAt statement. What happens and why?
- iv. Restore the gluLookAt statement and change the value of its arguments. See what happens. For example, try the following arguments:

- `gluLookAt(0.0, 0.0, 0.0, 2.0, 2.0, 2.0, 0.0, 1.0, 0.0);`
- `gluLookAt(2.0, 0.0, 2.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0);`
- `gluLookAt(2.0, 0.0, 2.0, 0.0, 1.0, 0.0, 0.0, 1.0, 0.0);`

h) gluAndGlutObjets.c: Modify the idle function to understand and change the spinning behaviour of the objects.

i) robotArm.c: Attach a “hand” at the end of the upper arm, using the gluCylinder function.

**3. If you have some time left, you can start building the parts of a 3D articulated object as part of your coursework.**

**Refer to the Labs4&5CWPart1.pdf document on Blackboard.**