University of Southampton	***School of Electronics and Computer Science***	***Coursework (2 of 4) Instructions***
Module: COMP 3004	Title: Principles of Computer Graphics.	Lecturer: Dr. J N Carter
Deadline: see below	Feedback: In laboratory	Weighting: 10%

Instructions

Introduction

Coursework 2 is lab based, where everybody will have a chance to implement different algorithms that could form part of a 3D *game* on Raspberry Pi's.
Coursework 2 will be from 10.00 to 13.00 in the Level 2 Electronics Laboratory in Building 59 (Zepler Building)

Lab dates are given in the table below. Your schedule is available on the web.

Dates
7 March 2018
14 March 2018
18 April 2018
25 April 2018

Deliverables.

Some guidance as to preparation and the code to be used will be published before the end of the week prior to the first lab. The exact details and the problem to solve will not be made available till the lab starts.

You are required to complete a short quiz relating to the work you have carried out. Your marks will be based on this and your progress.

Alternative submission, missing a lab

If you miss the lab session for any legitimate reason, but still want to complete the coursework without re-scheduling then you can submit a PC version of the exercise. Please contact the module leader (`jnc@ecs.soton.ac.uk`) before doing this. Since the solutions to the basic problems are freely available on the WWW after the last labs, PC submissions should be a functionally correct version of the basic problems, plus a working implementation of at least one of the advanced suggestions.

This should be submitted by posting on the internet on the Monday following the missed lab session, and e-mailing a link to the module leader. Please put an empty file called `pc.code` in the zip file so that it is clear what you have submitted.

If you need to reschedule your lab session, contact the module leader.

General Description

In this exercise you will be asked to

- Code an algorithm for a constant colour triangle,
- Code for Shaded triangles,
- Code for a Z-buffer.
- Code for a sphere built of triangles (from Coursework 1)

Each week you will code different algorithms, some that have been covered in lectures and some that may require reading ahead in the notes or the module text.

You will have access to Windows PC's running Windows 10 and Raspberry Pi 2 computers, together with all necessary software development tools. As I don't want to add to your already high coursework loads, these labs will be organized in a *just in time* format. The exact documentation and code for each week's lab will be published no more than a week ahead.

As preparation you are to look in module notes and module text. You should spend no more than an hour on this. In each lab you will be able to slot your code into an existing framework so that you will have a completed Raspberry Pi application demonstrating your work at the end of the day.

You will have the source code for my graphics library called Rational Graphics Library or RGL for short.

- This is based on SDL and provides basic tools for 2-D and 3-D drawing.
    - Polygon drawing using the XY and Edge Tracking Algorithms.
- A sample application to spin a wire frame cube I provided.
- The code contains place holders classes for the following:
    - Drawing a wire frame triangle (completed)
    - Rasterizing a triangle with a constant colour.
    - Rasterizing a triangle with interpolated colour.
    - Rasterizing a triangle with a interpolated colour plus a Z-buffer.
    - Drawing a shaded cube

Submission

You should complete the assessment test before you leave the laboratory and make sure the the laboratory supervisors have been given them. Please make sure that your names are on the sheets. Tests submitted at other times will not be marked without prior agreement.

Relevant Learning Outcomes (LOs)

1. Become familiar with the SDL/RGL libraries, and their application at the pixel level.
2. Demonstrate the ability to render triangles in 2-D with colour shading.
3. Show you understanding of three dimensional graphics and the coordinate transforms necessary to render a 3-D shape on a screen.
4. Demonstrate how three basic principles can be utilised to render a complex 3-D object, a cube.

Marking Scheme

Criterion	*Description*	*Session*	*LOs*	*Total*
Basic Shaders	Understanding the shader structure in RGL.	1	1	1
Shading	Implement a triangle Rasteriser using interpolation.	1	2	3
Back Face Culling	Don't draw triangle if it faces away from camera.	1	3	1
Shade Colour on a cube.	Shade a variable colour across the face of a cube.	2	2	2
Shading with Z buffer.	Interpolated Rasteriser using a Z Buffer, render a cube.	2	3	3
Light & colour	Render a shaded sphere (optional).	2	4	None
Coursework 2 is marked at the end of the lab and you are given the marked up version of the lab quiz form. No marks are recorded. Rather it is up to you to scan the form and submit it via Hand-in. This provides a permanent record of your work.				

Late submissions will be penalised at 10% per working day. No work can be accepted after feedback has been given.
You should expect to spend up to 8 hours on this assignment, that is 2 hours preparation and 6 hours lab time. Please note the University regulations regarding academic integrity.