COMP3421 and COMP9415 Computer Graphics 15s2 Course Outline

Aims

The aim of this course is to teach the fundamental algorithms underlying computer graphics in both 2D and 3D.

Topics covered include:

- The OpenGL API for 3D rendering
- Drawing in 2D: lines and polygons
- The difference between world, camera and screen coordinate systems
- Affine transformations and homogeneous coordinates
- The OpenGL graphics pipeline
- 3D Meshes
- Clipping in 2D and 3D
- Smooth Shading
- Illumination and shadows
- Hidden surface removal
- Depth buffers
- Textures
- Converting lines and polygons to raster images
- Anti-aliasing
- Bezier curves, B-splines, NURBS
- Representing colour
- Ray tracing
- Shader programming in GLSL

Pre-requisites

The formal prerequisite for this subject is COMP2911

You are expected to have a working knowledge of:

- Programming in Java
- Object-oriented design with interfaces and patterns
- Basic linear algebra: matrices and vectors

Learning outcomes

Knowledge and Understanding

After successfully completing this course you should have knowledge and understanding of

- **modeling:** the mathematical representation and computer implementation of lines, curves, surfaces, and transformations
- **rendering:** the mathematics of projection, hidden surface removal and local and global illumination; the computer implementation of this in the graphics pipeline
- how modeling and rendering work together in graphics hardware and software

Practical Skills

After successfully completing this course you should be able to use OpenGL to write interactive computer to manipulate and render complex 3D scenes.

Staff

Lecturer in Charge:
Angela Finlayson
angf@cse.unsw.edu.au

Tutors:
Rob Everest
robertce@cse.unsw.edu.au

Oliver Tan
otan@cse.unsw.edu.au

Adam Chyb
ajch433@cse.unsw.edu.au

Lectures

	Week Day	Start Time	End Time	Location	Weeks
	Wednesday	15:00	18:00	Central Lecture Block 6 (K-E19-103)	1 to 12

Labs

In **week 2** you will have a **lab** instead of your tutorial. In your normal tutorial time you will go to the **piano** lab for **week 2 only.** This is your chance to get help setting up jogl on your laptops too if you need it, so bring them along as well.

Tutorial Classes

	Week Day	Start Time	End Time	Location	Weeks
M13A	Monday	13:00	14:00	Goldstein G02 (K-D16-G02)	3 to 13
M14A	Monday	14:00	15:00	Goldstein G07 (K-D16-G07)	3 to 13
W12A	Wednesday	12:00	13:00	Quadrangle G031 (K-E15-G031)	3 to 13
W13A	Wednesday	13:00	14:00	Quadrangle G031 (K-E15-G031)	3 to 13
W18A	Wednesday	18:00	19:00	Quadrangle 1046 (K-E15-1046)	3 to 13
H14A	Thursday	14:00	15:00	Quadrangle G041 (K-E15-G041)	3 to 13
H15A	Thursday	15:00	16:00	Quadrangle G041 (K-E15-G041)	3 to 13
H16A	Thursday	16:00	17:00	Quadrangle G042 (K-E15-G042)	3 to 13

Resources

The textbook is Computer Graphics using OpenGL by FS Hill. It doesn't matter whether you have the 3rd or 2nd edition.

The text book uses C++ for its example programs, but you should have no trouble understanding them if you are familiar with Java or C.

For lecture examples and assignments will be using JOGL, a Java interface to the native OpenGL libraries.

Consultation

Consultation hours for this subject are Tue 2:30-3:30pm in K17-G02

Assessment

Assessment will be based on 2 assignments and a final exam and is subject to scaling. Contributions are as follows:

Task	Value
Assignment 1	15%
Assignment 2	25%
Exam	60%

Assignment 1

Released: Week 2

Due: End of week 5

Covers:2D drawing and transformations in OpenGL

Assignment 2

Released: Week 7

Due: Milestone 1 due end of week 9, final submission due end of week 12. Demo in week 13

Covers:3D drawing and transformations in OpenGL

Exam

There will be a two hour final exam at a time to be arranged, covering the entire syllabus.

The exam will be open book. Calculators and rulers will also be allowed into the exam.

Plagiarism

Plagiarism (taking the ideas or words of others and passing them off as your own) will not be tolerated. See the UNSW Statement on Plagiarism at: https://my.unsw.edu.au/student/atoz/Plagiarism.html

Teaching Rationale and Strategies

Computer Graphics is fun. You can use it to make cool games and amazing special effects in movies. So learning it should be fun.

Computer Graphics is hard. The reason why the world looks like it does is because there are gazillions of photons bouncing around between all the objects. Simulating all the photons and the properties of the objects is too hard to do, so we have to come up with mathematical approximations and clever algorithms.

So you are going to have to work hard to understand the maths and the algorithms. But then you get the reward of making pretty pictures. Which is even more fun because you have earned it.

Course Feedback and Improvement

Student feedback on this course will be obtained via electronic survey at the end of session and will be used to make continual improvements to the course. Students are also encouraged to provide informal feedback during the session and to let the lecturer in charge know of any problems as soon as they arise. Suggestions will be listened to very openly, positively, constructively and thankfully, and every reasonable effort will be made to address them.

Feedback from the previous session has indicated that some students felt there was not enough emphasis on more modern opengl, such as shaders and retained mode. This session we will be spending more time on these topics and introducing them earlier.