COMP3011 Computer Graphics

Spring 2025

Assessment 2

Introduction

This coursework will test your understanding of the concepts of 3D computer graphics taught during the course, and your ability to create, implement and display a 3D scene of your choice. You have been provided with APIs to use during the course, e.g. camera.h and tiny_obj_loader, and you can use them in your coursework. **You cannot use any other API**.

You are required to write a report outlining how you have met the requirements. The report should be no more than 6 pages and should contain screenshots of your program. **The report is compulsory. Failure to submit a report will result in a zero mark for the assessment.**

You are also required to demonstrate your running program to a member of the teaching team. Your demo will need to highlight your coursework features, and you will be required to answer questions about your program and its implementation. This will be an opportunity to show off your knowledge and skills in computer graphics. **The demonstrations are compulsory. Failure to attend the demonstration will result in a zero mark for the assessment.** You will be provided the location in A32 and a time slot for the demonstration in due time.

Submission

You need to submit two separate files to Moodle:

1. A zip file containing your visual studio solution and all required files. Before submitting your code, test it on an A32 lab PC. Marks will be lost for programs that do not compile or have issues with linking to resources. The marker will download your submitted zip file, unzip it to a directory on a school PC, open the Visual Studio .sln file, click the Run Button, and your project should compile and run without any further interventions.
   You must submit a zip file, a link to a file location will not be accepted.
2. Your report. You must fill in the provided report sheet.

Technical Requirements

The technical requirements, as covered in the lectures and lab sheets, are as follows:

TR1 – General 3D Graphics Programming

You need to demonstrate the essentials of 3D graphics programming including window creation, handling input, using depth test and antialiasing, specifying VAOs, correct use of projection, and a fragment and vertex shader.

TR2 – 3D Modelling

See the example projects on Moodle. You **must** include both:

1. *A procedurally generated object*. Marks for this object will be based on the complexity of the object.
2. *An OBJ file object*. You **must** provide OBJ files in your submission.

TR3 - 3D Transformations & Animation

For high marks you need to demonstrate that you can apply transformations, e.g. rotate, translate, and scale, to achieve a desired effect, for example automatically transforming an object through the scene over time. This must be separate from the moving camera.

TR4 - Cameras

Implement at least one camera. Marks will be awarded for the complexity of the camera you implement, including animated or multiple cameras.

TR5 - Textures

The mark will be proportionate to the complexity of the mapping, not the complexity of the texture. The absolute minimum is to map a texture to a triangle. Bonus available for using mipmaps.

TR6 - Lighting

Create a light by specifying its location and properties. You will receive marks for each type of light you include and bonus will be given for combining lights.

TR7 - Shadow

Dynamic shadows need to look accurate according to the lights and geometry in the scene.

TR8 – An interactive object

You need to have an interactive object which responds to user input. This needs to be separate from camera control. Bonus available for more imaginative interaction.

TR9 – Curves

Implement Bezier curves and use a curve with at least 3 control points. Bonus for using curves with more control points and imaginative use of the curves.

TR10 – Transparency

Include a transparent surface in your scene which is blended with surfaces behind. Bonus available for correctly rendering overlapping transparent surfaces.

Research & Development

Marks are available for you to program something I haven't given a lecture on.

Marking Scheme

Note the highlighted technical requirements apply to each of the 2 object types described in TR2.

COMPONENT	**WEIGHT**
Report	1%
Tidy Code	1%
Demo	1%
TR1 – General 3D Graphics Programming	20%
TR2 – 3D Modelling	20%
TR3 – 3D Transformations	3%
TR3 – Animation	3%
TR4 – Cameras	3%
TR5 – Textures	2%
TR6 – Lighting	4%
TR7 – Shadow	5%
TR8 – An interactive object	1%
TR9 – Curves	3%
TR10 – Transparency	3%
R&D	30%
TOTAL	**100%**

You will be awarded marks for using your implementations from the lab exercises.
You can receive more marks if you implement an idea from the lectures which is not in the lab exercises.
You will be penalised for the following:

1. If your program requires any changes to make it run.
2. If your program crashes.
3. If you use any API not provided during the course or programmed yourself.
4. If there are any apparent visual errors in the rendered image.
5. If your program obviously does not use concepts described in the course, e.g. algorithms, unless you adequately justify using a different algorithm. For example, you may use an algorithm for curves which is different to Bezier curves, which is fine if you justify it.
6. If your program has a framerate which is not interactive.
7. If your program requires more than 60 seconds to load, after compiling and executing.
8. If you submit a report sheet with your own formatting.