School of Design and Informatics

Session 2020/21

Module Code: **CMP203**

Module Title: **Graphics Programming**

Module Deliverer: **Dr Paul Robertson**

Unit of Assessment: **Unit 1 of the module assessment – 100%**

Submission date: **Tuesday 5ᵗʰ January 2021.**

Suggested Feedback Return Date: **25ᵗʰ January 2021.**

Feedback Type: **Feedback will be provided on My Learning Space.**

Assessment overview

This coursework comprises two parts; a developed application and an accompanying report. To aid in your coursework a starter application is provided via GitHub Classroom and this also functions for the submission of the code. The free private repo and starter application can be accessed through the link provided on My Learning Space. This works similar to the lab projects. This will require a GitHub account. **Clone** the provided starter repository. For submission make sure all changes are **committed and pushed** by the submission deadline. On the submission deadline all repos will be automatically downloaded.

Task 1: Build a Graphics Application

Design and develop a 3D graphics application and scene that exhibits key techniques in graphics programming. The application should demonstrate your ability to use the OpenGL API in a structured and efficient manner. The 3D scene constructed can be of your own design or a limited re-creation of a scene from a game or film. Note, that I do NOT expect you to create the gameplay from the game only a mostly static scene with some interactivity is expected. A single 3D object on a black background is insufficient. Multiple objects that complement each other arranged into a 3D scene representing a recognisable scene are required.

Only the OpenGL API may be used to construct your rendering system. Third party game engines cannot be used. However, if you wish to use direct input for Input or FMOD for sound, this is permissible.

A good coursework should include all the following features:

- Geometry
    - The scene **must** contain user created geometry, not just loaded models.
    - This geometry **must** correctly lit and textured.
    - Additionally, the geometry could demonstration transparency and use of depth sorting.
    - Demonstration of procedurally generated shapes/primitives. Generated shapes should be lit and textured correctly.
    - User created geometry should demonstrate use of vertex arrays (beyond model loading).
- Lighting
    - The scene **must** show lighting from multiple lights of different types, colours and some animated.
    - Consider how your scene will use mix of light sources.
    - These light sources could be animated or made user controllable.
- Camera and Interaction
    - The scene **must** contain a working camera. The user must be able to manipulate the view through using the mouse and keyboard to control the camera.
    - Additionally, you should provide multiple cameras each with a different control scheme such as limited controls, procedurally controlled views, tracking controls or different camera types (such as 3rd person).
    - User interaction (controlling objects in the scene other than the camera).
- Hierarchical Modelling
    - A clear example of Hierarchical modelling and animation through hierarchical means (with use of the matrix stack).
- Advance features
    - Demonstration of shadows and use of the stencil buffer.
- A wireframe mode.
- The application code should be carefully designed and constructed showing appropriate use of classes and well commented.

This exercise is designed to let you demonstrate your knowledge of OpenGL and graphics programming concepts covered in class.

The assignment is designed to be very wide in scope so there is plenty room for creativity. If you are concerned your idea does not meet the requirements, please consult the module tutor to ensure the idea is appropriate for submission.

ENSURE YOUR APPLICATION WORKS ON THE LAB COMPUTERS

Task 2: Report

The created application should be accompanied by a 2,500-word report explaining, in detail, how the scene was constructed and how your work meets the coursework requirements. The report should contain:

1. A short overview of the scene
2. Any relevant user controls for the application
3. A detailed explanation of the major elements of the scene and how they are constructed/work. Highlighting how they address the coursework brief. The explanation should include screenshots and pseudocode to aid the explanation.
    a. For example, for the procedural shape generation explain what shapes were generated, how they are generated, how texture coordinates and how the normals are calculated. Providing screenshots of the shapes and code explanation of the generation and storage.
4. References of any textures/models, tutorials, etc. used. References must be Abertay Harvard Standard formatting.

A basic template for the report is provided on My Learning Space.

Submission

Submit electronically via GitHub and My Learning Space by Tuesday 5ᵗʰ January 2021.

- The code/project solution is to be submitted via the provided GitHub Repository. This will be automatically downloaded on the date of submission. You must make the appropriate commit and push before that deadline.
- A standalone executable version of the application and report should be submitted to My Learning Space. These files should be contained within a zip file for uploading, using the following folder structure:
    o A folder titled "exe" containing the executable file and any textures/models required for the project to run standalone.
    o And a PDF version of your report.
    o Zip files should be sensibly titled e.g. CMP203_JohnSmith.zip.

All submissions must be uploaded to the appropriate location within the My Learning Space. You will be able to make multiple submissions (in case of errors) but only the last submission will be marked.

Marking scheme

Literal Grade	Evaluative Descriptor	This Assessment
A+	Excellent overall. • Demonstrates an excellent grasp of the subject matter. • Excellent capacity for original and creative enquiry. • Excellent ability to critically evaluate, analyse, synthesise and integrate complex information. • Excellent communication skills. In addition, exceptional in at least one of the above.	
A	Excellent overall. • Demonstrates an excellent grasp of the subject matter. • Excellent capacity for original and creative enquiry. • Excellent ability to critically evaluate, analyse, synthesise and integrate complex information. Excellent communication skills.	A meticulously constructed application demonstrating all of the graphical programming techniques outlined in the specification. Extensive use of lighting, complex user-created geometry, several complex cameras and demonstration of advance techniques such as complex shadowing and stencil buffer use. Code is well structured and well commented with good use of additional classes. Report is well detailed showing clear and detailed understanding of the work completed.
B+	Very good overall. • Demonstrates a very good grasp of the subject matter. • Very good capacity for original and creative enquiry. • Very good ability to critically evaluate, analyse, synthesise and integrate complex information. • Very good communication skills. In addition, excellent in at least one of the above but overall performance deemed to be very good.	
B	Very good overall. • Demonstrates a very good grasp of the subject matter. • Very good capacity for original and creative enquiry. • Very good ability to critically evaluate, analyse, synthesise and integrate complex information. Very good communication skills.	A well-constructed application demonstrating many of the graphical programming techniques outline in the brief. One or two techniques missing or lacking in complexity. Code is well structured and well commented with good use of additional classes. Report is well detailed, showing very good understanding of the subject.
C+	Good overall. • Demonstrates a good grasp of the subject matter.	

	• Good capacity for original and creative enquiry. • Good ability to critically evaluate, analyse, synthesise and integrate complex information. • Good communication skills In addition, very good in at least one of the above but overall performance deemed to be good.	
C	Good overall. • Demonstrates a good grasp of the subject matter. • Good capacity for original and creative enquiry. • Good ability to critically evaluate, analyse, synthesise and integrate complex information. Good communication skills	The constructed application demonstrates some of the graphical programming techniques discussed in class. Shows limited work outside that covered in class or techniques missing or lacking complexity. Code shows some good structure and some commenting. Report shows good understanding of the work completed but requires a more detailed description of the work.
D+	Satisfactory overall. • Demonstrates a satisfactory grasp of the subject matter but limited grasp in some areas • Satisfactory capacity for original and creative enquiry. • Satisfactory ability to critically evaluate, analyse, synthesise and integrate information. Satisfactory communication skills	
D	Adequate. Achievement of all threshold standards but grasp of some subject areas and graduate attribute development may be more limited.	The application meets the minimum requirements. Lacking many of the suggested graphical programming techniques. Possibly showing little beyond class examples given. Report shows basic understanding of the subject.
MF	Marginal fail. Performance just below the threshold standard. A reasonable expectation that a pass is achievable by reassessment without the need to repeat the module.	A very simple application that fails to meet the minimum requirements. Poor or no report of work completed.
F	Performance well below the threshold level. Some limited evidence of achievement of the outcomes.	No working application or little evidence of work. Little or no report submitted. Little or no indication of understanding of subject.
NS	No assessments submitted or no evidence of achievement of the outcomes.	