# COMP3015 GAME GRAPHICS PIPELINES

**20 CREDIT MODULE / 100% COURSEWORK SUBMISSION**

**MODULE LEADER:** **Ji-Jian Chin**

## Aims

- To utilize and gain insights into industry standard frameworks and approaches.
- To apply optimisation techniques for computer graphics

## ASSESSED LEARNING OUTCOME:

1. Design, conceptualise and implement a working prototype with clearly defined features and optimised performance.
2. Deliver a tool from the primary feature set with end user documentation released on an appropriate public platform

## OVERVIEW

COMP3015 module introduces concepts and programming techniques for working with shaders (little programs for coding the graphic card) and an understanding of a Computer Graphics Pipeline, using graphical API's like OpenGL. The focus will be on OpenGL Shading Language (GLSL), which is a high-level shading language with a syntax based on C. Throughout the module, you will learn to create custom sheading models, techniques to optimise the rendering pipeline and various special effects used in game development.

Lectures, seminars, and labs are part of the module delivery each week. The emphasis is on the theoretical side during the lecture, followed by practical examples during the seminars. The labs are reserved for students to implement and explore, recently taught concepts.

Submissions for this module are 100% coursework comprising of two parts, C1 - Initial Prototype - 30% and C2 Optimised Developer Tool with documentation - 70% as explained bellow:

**C1 Initial Prototype - 30%**: An OpenGL project utilizing a custom shading model implemented in GLSL, showcasing a textured based technique. A link to the repo used for developing the prototype and a video presenting your approach and implementation.

**C2 Optimised Developer Tool with documentation - 70%**: An OpenGL project, with the focus on GLSL implementation of at least 3 basic shaders techniques taught in the module or an indepth technical implementation of one shader technique. Research and produce a short industry standard portfolio piece (GitHub page) and a video presenting your approach and implementation.

## CONTACT

Module Leader: Ji-Jian Chin (ji-jian.chin@plymouth.ac.uk)

## MODULE DELIVERY

**Delivery format**: Lecture + Seminar (2 hours) – **Mondays** from **16:00 to 18:00**,
Lab (2 hours) – **Thursdays** from **9:00 to 11:00** and a repeat **Thursdays** from **11:00 to 13:00** each week for a total of 12 weeks

**Delivery location**: Lectures: PSQ102 and
Labs: SMB200

**Delivery staff**: Ji-Jian Chin

**Important dates**: **Reading week**: Week 34, 20/03/2023. Lecture session is for reading and we use the labs for feedback and catch up.

**Submission1: Week 35, 27/03/2023, 15:00 – C1 Initial Prototype, DLE (summative)**

**Easter Break**: Week 36 – 38, 03/04/2023 – 21/04/2023
**Early May Bank Holiday**: Week 40, 01/05/2023 – No Lecture on Monday, moved to Tuesday
**Coronation Bank Holiday/Reading Week**: Week 41, 08/05/2023 – No Lecture on Monday, moved to Tuesday but for reading week and we use the labs for feedback and catch up.

**Submission: Week 17, 17/05/2023, 15:00 – C2 Optimised Developer Tool with documentation, DLE (summative)**

NB: For a 20 credit module there will be on average 50hrs of taught content and staff led activities with 150hrs of independent study - students are expected to work through exercises, read and assimilate provided resource materials and prepare for the next session, final assessment will include elements from further study materials, extending taught materials. It is essential you apply your study skills and provide evidence of this additional work.

## ASSIGNMENT SPECIFICATION

### C1: Initial Prototype 30%

**Prototype** - An OpenGL project utilizing a custom shading model implemented in GLSL, with custom lighting set up, showcasing a textured based technique.

Objective is to implement a custom shading model using GLSL that makes use of the current techniques and approaches used in the industry. The shading model should work with at least one textured 3D model and one light.

There are different shading models available: Flat shading, Gouraud shading, Phong shading, Blinn-Phong, etc. You should think of a combination between the shading model and the number of lights or the type of lights used.

The textured model can have a simple texture applied or a more complex texturing approach to create a unique combination together with the shading model.

Creativity in scene set up plays an important role. What's the best approach to showcase your graphics techniques? Scene complexity or/and "intentional visuals" (trying to replicate the look and feel of your favourite game) approach will help you focus your development.

**Video** – YouTube video containing a summary of the developed project with a recommended length between 5 to 10 minutes. Student should cover elements that make the prototype unique,

covering the development process from the initial idea to the final product. Focus the discussion on the shading model, lighting used and the texture manipulation for the mesh. Shader code review should be also part of the video together with various problems solved during development. Indicative resolution and compression settings for this video are shown below.

| | |
|---|---|
| File-Type: | MP4 |
| Resolution: | 1440p (2K) |
| Framerate: | 30 |
| Video Bitrate: | 16 MBS |
| Audio Bitrate: | Mono – 128 kpbs, Stereo – 384 kpbs |
| Compression: | H.264 |

**C1 – Deliverables and deadlines**

Submit to DLE a single zip folder named “Project.zip” containing:

.

- Your VS project based on the template given to you by me at the beginning of the module. Please make sure you remove the “Debug”, “ipch” or”.sdf” files or folder. Your project should be a couple of MB’s anything over 150MB means you didn’t remove these files. There is a limit on the DLE of 150MB so not removing these files might stop you from submitting your assignment.
- An executable version of the project which does not rely on VS • Include any resources such as extra assets that your project needs.
- A brief write-up in markdown format, describing:
    - Which version Visual Studio and Operating System you used to test/write the code locally?
    - How does it work?
    - How does your code fit together and how should a programmer navigate it (not intended to be an exercise in formally documenting the code)?
    - Anything else which will help us to understand how your prototype works.
    - A Link to the unlisted YouTube Video

| Part | Description | Deadline | Percentage |
|---|---|---|---|
| C1 | Initial Prototype<br><br>– Formal submission on DLE | Week 35, 27/03/2023, 15:00 | 30% |

**Marking rubric**

This is an indicative marking rubric to give you an indication of your project's level:

| Category | Under 40 | 40-50 | 50-60 | 60-70 | Over 70 |
|---|---|---|---|---|---|
| Initial prototype | No submission<br><br>Software doesn't compile<br><br>Cannot explain the shader code.<br><br>Empty project with just a shader or project with no shader.<br><br>There's no scene set up, no model.<br><br>Any plagiarism in any form, including your own previous work. | Basic prototype with minimum implementation of a shading technique, simple light implementation, and a very minimalistic texturing.<br><br>The video is very basic with minimum explanation of the code.<br><br>There's not enough evidence to show the student understood the concepts.<br><br>The scene set up is very basic with typically one or two 3D objects used. | Prototype with a set of average complexity in shading techniques, one or more lights set up and texturing is basic.<br><br>The video has some explanations about code but not in a lot of depth.<br><br>Evidence of understanding of the basic concepts.<br><br>The scene set up is average with some 3D elements but static. | Prototype with a complex shading technique implementation, complex light set up and model texturing beyond basic levels.<br><br>The video has good explanations about code and nicely presented.<br><br>Strong evidence of understanding the concepts, even the most challenging ones.<br><br>The scene set up shows creativity with dynamic elements and complementary visual elements. | Great prototype implementation with one or more complex shading techniques. Tidy code, very efficient neatly commented and explained.<br><br>The video has in-depth explanations of the approach, professionally presented.<br><br>Student clearly went beyond the taught content and done extensive work on their own with very clear understanding of the concepts.<br><br>The scene set up is intentional with creative ways of showcasing the implemented techniques. |

## C2: Optimised Developer Tool with Documentation 70%

**Optimised Developer Tool** - An OpenGL project utilizing at least 2 of the techniques in the following categories: Geometry shaders, Shadows, Noise, Particle system and animation, Image processing techniques, Physically based reflection model (PBR).

Objective is to create a project that showcases at least 2 of the above techniques combined into one efficient render pipeline. It is perfectly ok to use elements from the first submission (C1) as a start, but you need to show clear contribution beside that initial prototype. The outcome should be a portfolio piece that you can highlight your skills as a graphics programmer using GLSL. This application can be used for potential employment therefore attention to detail is crucial. Follow the traditional game loop to show your work in the context of a game and make sure the shader code implementation is tidy and easy to read.

You may use any from any sources (i.e. “gamedev”, Nvidia, Shadertoy) of program code which you can adapt. But you must tell us what you started with and what you have done to it. If this is not done you risk failing the assignment. You can also use a wider range of 3rd party libraries this time around. You could also use a physics library such as a Bullet or PhysX.

There must also be a substantial contribution from yourself. For example, just taking an existing shader and changing the names is not enough. You need to have a combination of at least 2 techniques. You should also consider recycling parts of the first assignment but don’t submit the entire project with no modifications, that is a fail.

And finally, if I have any doubts about the origin of the code, I might ask you questions about the program code you submitted.

**YouTube video** – YouTube video containing a summary of the developed project with a length between 5 to 10 minutes. Student should cover elements that make the prototype unique, covering the development process from the initial idea to the final product. Focus the discussion on the shading model, lighting used and the texture manipulation for the mesh. Shader code review should be also part of the video together with various problems solved during development.

Indicative resolution and compression settings for this video are shown below.

| | |
|---|---|
| File-Type: | MP4 |
| Resolution: | 1440p (2K) |
| Framerate: | 30 |
| Video Bitrate: | 16 MBS |
| Audio Bitrate: | Mono – 128 kpbs, Stereo – 384 kpbs |
| Compression: | H.264 |

**C2 – Deliverables and deadlines**

| Part | Description | Deadline | Percentage |
|---|---|---|---|
| C2 | Optimised Developer Tool with Documentation<br><br>– Formal submission on DLE | Week 42, 17/05/2023, 15:00 | 70% |

Create a public GitHub repository for your submission which is documented with a markdown file explaining:

- How does the user interact with your executable? How do you open and control the software you wrote (exe file)?
- How does the program code work? How do the classes and functions fit together and who does what?
- What makes your shader program special and how does it compare to similar things? (Where did you get the idea from? What did you start with? How did you make yours unique? Did you start with a given shader/project?)
- Include a link to a video report that details the above but also goes into more detail on:
    - Anything else which will help us understand how your shader works.
    - Are there any software engineering issues, such as the trade-off between performance and good practice?
    - A (brief) evaluation of what you think you have achieved, and what (if anything) you would do differently, knowing what you now know. Don't be modest but be realistic!

Create a single ".zip" archive containing the whole Visual Studio project (not just the executable), with the solution we should open clearly indicated (put that in the write-up below). Don't forget to include any other resources required; meshes, sounds etc. Submit your archive via the DLE electronic submission system.:

- Your VS project based on the template given to you by me at the beginning of the module. Please make sure you remove the "Debug", "ipch" or ".sdf" files or folder. Your project should be a couple of MB's anything over 150MB means you didn't remove these files. There is a limit on the DLE of 150MB so not removing these files might stop you from submitting your assignment.
- An executable version of the project which does not rely on VS.
- Include any resources such as extra assets that your project needs.
- A brief write-up in markdown format, describing:
    - Which version Visual Studio and Operating System you used to test/write the code locally?
    - How does it work?
    - What makes your shader program special and how does it compare to similar things? (Where did you get the idea from? What did you start with? How did you make yours unique? Did you start with a given shader/project?)
    - Anything else which will help us to understand how your prototype works.
    - A link to the GitHub repo you created for public use
    - A Link to the unlisted YouTube Video

**Marking rubric**

This is an indicative marking rubric to give you an indication of your project's level:

| Category | Under 40 | 40-50 | 50-60 | 60-70 | Over 70 |
|---|---|---|---|---|---|
| Initial prototype | No submission<br><br>Software doesn't compile<br><br>Cannot explain the shader code.<br><br>Empty project with just a shader or project with no shader.<br><br>No public GitHub.<br><br>Any plagiarism in any form, including your own previous work. | Basic prototype with one shader technique.<br><br>The video is very basic with minimum explanation of the code.<br><br>Minimum evidence of student contribution and understanding of the basic concepts.<br><br>Very Basic implementation of a public GitHub.<br><br>The scene set up is very basic with typically one or two 3D objects used. | Prototype with at least couple of shading techniques.<br><br>The video has some explanations about code but not in depth.<br><br>Evidence of understanding of the basic concepts but in no depth.<br><br>GitHub is available with ok documentation and links to the video.<br><br>The scene set up is average with some 3D elements but static. | Prototype is well built with good shading techniques applied.<br><br>The video has good explanations about code.<br><br>Strong evidence of understanding the concepts, even the most challenging ones.<br><br>GitHub is presented well with good explanations and in depth set up instructions.<br><br>The scene set up shows creativity with dynamic elements and complementary visual elements. | Great prototype implementation with tidy code and very efficient. Great comments in the code.<br><br>The video has in-depth explanations of the approach, professionally presented.<br><br>Student clearly went beyond the taught content and done extensive work on their own with very clear understanding of the concepts.<br><br>A very professional GitHub presentation ready for public access.<br><br>The scene set up is intentional with creative ways of showcasing the implemented techniques. |

## COURSEWORK SUBMISSION

We require a digital submission uploaded to the module DLE site, allow sufficient time to upload project work and documentation. If your submission exceeds file size supported by the DLE (150MB) please set up a folder on your OneDrive account with module code: 'COMP3015' set permissions to allow staff access – do not modify or remove this folder until results are confirmed. If your DLE submission documentation includes additional URL links, please ensure you have set the permissions to public and test all from an anonymous browser window.

**Please refer to all the lecture content & further study resources on the DLE.**

## GENERAL GUIDANCE

For this assignment you may be using information from differing sources:

- Books, journal articles
- Course/module materials
- Websites
- Existing Open-Source Projects

It is **very important** for you to note that this assignment is an **individual effort**. It **should clearly highlight the contributions from the student** and the use of external resources or an initial starting project.

Thus, do not simply copy existing sources, i.e., other students work, interspersed with a few lines of code or words of your own. This is paraphrasing, and it is not encouraged, it is not likely to get you a good mark and, in some cases, it could be seen as plagiarism. In a similar vein, do not simply copy material from elsewhere without citing it properly.

Please see the Plagiarism section below. If you have any doubt as to what constitutes '**an individual effort and in your own words**' then either see your student handbook or see me.

**Referral**

Please note that if you claim for non-submission and are offered a referral, you will be required to complete a NEW piece of work for the module. The new piece of work will assess that you have met the learning outcomes for the module but in a way that will be different to the original set piece. The referral is not a repeat or extension of the original coursework.

Carrying out a new piece of work means you will not be able to keep the marks already gained during the module. Eg: if you pass the set exercises (CW1) but do not submit the main work (CW2) AND are offered a referral, you will not keep the set exercises grade.

**Extenuating Circumstances**

There may be a time during this module where you experience a serious situation which has a significant impact on your ability to complete the assessments. The definition of these can be found in the University Policy on Extenuating Circumstances here: https://www.plymouth.ac.uk/uploads/production/document/path/15/15317/Extenuating_Circumstances_Policy_and_Procedures.pdf

**Plagiarism**

All your work must be of your own words. You must use references for your sources regardless how you acquired them. Where you wish to use quotations, these must be a very minor part of your overall work.

To copy another person's work is viewed as plagiarism and is not allowed. Any issues of plagiarism and any form of academic dishonesty are treated very seriously. All your work must be your own and other sources must be identified as being theirs, not yours. The copying of another persons' work could result in a penalty being invoked.

Further information on plagiarism policy can be found here:

Plagiarism: https://www.plymouth.ac.uk/student-life/your-studies/essentialinformation/regulations/plagiarism

Examination Offences: https://www.plymouth.ac.uk/student-life/your-studies/essentialinformation/exams/exam-rules-and-regulations/examination-offences

Turnitin (http://www.turnitinuk.com/) is an Internet-based 'originality checking tool' which allows documents to be compared with content on the Internet, in journals and in an archive of previously submitted works. It can help to detect unintentional or deliberate plagiarism.

It is a formative tool that makes it easy for students to review their citations and referencing as an aid to learning good academic practice. Turnitin produces an 'originality report' to help guide you. To learn more about Turnitin go to:
https://guides.turnitin.com/01_Manuals_and_Guides/Student/Student_User_Manual

**Referencing**

The University of Plymouth Library has produced an online support referencing guide which is available here: http://plymouth.libguides.com/referencing.

Another recommended referencing resource is Cite Them Right Online; this is an online resource which provides you with specific guidance about how to reference lots of different types of materials.

The Learn Higher Network has also provided a number of documents to support students with referencing:
References and Bibliographies Booklet:
http://www.learnhigher.ac.uk/writing-for-university/referencing/references-and-bibliographiesbooklet/
Checking your assignments' references:
http://www.learnhigher.ac.uk/writing-for-university/academic-writing/checking-your-assigmentsreferences/