

2022 Subject & Assessment Guide

Computer Graphics

10702NAT

Advanced Diploma of Professional Game Development

Programming

Table of Contents

Computer Graphics	3
Units of Competency	3
Overall Learning Outcomes.....	3
Subject Description	3
Industry Relevance.....	4
Assumed Knowledge.....	4
Subject Textbooks	4
Assessment Criteria	5
Assessment Description	5
Software.....	9
Core.....	9
References and additional material.....	9

Computer Graphics

Units of Competency

The units of competency that are covered in this subject are as follows:

[ICTICT427](#) – Identify, evaluate and apply current industry-specific technologies to meet organisational needs

Assessment processes and competency evidence requirements are described in the *Assessment Criteria* section below. If you have prior or other evidence against competency you should discuss this with your teacher.

Subject Overview

Graphics is an expansive domain of knowledge, tools and technology to achieve everything that eventually reaches our screens. This instruction module will cover techniques, libraries, and algorithms commonly used in modern real-time rendering. We will explore the relationship between and management strategies for vertex data, program shaders, and framebuffers.

Overall Learning Outcomes

The overall learning objectives for this subject are:

- Demonstrate an understanding of modern render pipelines on Graphics Processing Units (GPUs)
- Apply Industry standard rendering techniques for games, film and simulation
- Apply knowledge of industry standard game engine render pipelines

Subject Description

This subject is designed to teach you the **techniques** and **algorithms** used in **modern real-time rendering**.

Computer Graphics (CG) has a history dating back decades. From humble beginnings in the 1940's to the first mainstream use in movies in the 1970's, CG kept improving over the years from simple wireframe renderings to complete 3-dimensional (3-D) representation of everyday objects.

As techniques improved so did their application in film and video games, with games evolving from text-based experiences to 2-dimensional before the leap to real-time 3-D games was made.

Modern video games make use of advanced dedicated computer hardware to display amazingly life-like visuals, hardware which is now included with almost every mobile device today. With this subject you will learn how to take control of this hardware, making use of the **OpenGL** API to push the hardware to its limits.

Industry Relevance

Many games studios implement their own game engines while others make use of third-party solutions such as Unity3D and Unreal Engine. The skills taught within this subject would allow a student to fully understand the workings of either in-house or third-party game engines at any studio and to be able to create their own solutions. In addition, the techniques learned are the same as those used in related industries, such as film and simulation.

Assumed Knowledge

- Knowledge of C++ programming sufficient to create complex real-time applications
- Knowledge of basic vector and matrix mathematics for 3-D coordinate systems
- Knowledge of game engines

Subject Textbooks

Although not required, the following textbooks are recommended to aid in the completion of this subject:

- Wolff, D., 2018. **OpenGL 4 Shading Language Cookbook: Build High-quality, Real-time 3d Graphics With OpenGL 4.6, GLSL 4.6 And C++17**, 3rd Edition. Packt Publishing.
- Sellers, G., 2015. **Opengl Superbible: Comprehensive Tutorial And Reference (7th Edition)**. Addison-wesley Professional.
- Akenine-Möller, T., 2018. **Real-time Rendering, Fourth Edition**. A K Peters/CRC Press

Assessment Criteria

Assessment Description

Assessment Milestones

Please refer to your Class Schedule for actual dates on your campus

General Description

Throughout this subject you will be learning graphics programming skills that apply to games, film and simulations. You will be demonstrating your understanding through the development of two 3D applications: an **OpenGL application** showcasing the use of the GPU for low-level rendering, and an application produced using a **game or graphics engine** that showcases the use of advanced rendering techniques.

Presenting your applications, recording, and responding to feedback is a required part of this assessment.

3D OpenGL Application

On the resources page for this subject on Canvas (<https://aie.instructure.com/>) you will find a collection of models you can use in the implementation of a 3D scene constructed within your OpenGL application.

The **3D OpenGL application** must be developed using C++ and demonstrate the following techniques:

- Imported 3D models, rendered using custom GLSL shaders
- Texture mapping applied to at least one imported 3D model
- 3D lighting applied, using at least 2 light sources at a time

Note: you will not be assessed on how accurately you reproduce the scene, but rather on the application and demonstration of these techniques.

3D Game Engine-Developed Application

The **3D game engine-developed application** can be created using a 3D engine agreed upon by your teacher, such as Unreal Engine 4, Unity3D, or a custom 3D rendering engine, and must demonstrate the following techniques within a single scene:

- An interactive 2D GUI that the user can use to interact with the 3D world in some manner, such as change materials on an object, spawn effects, or toggle effects on/off.
- Custom materials created using the engine's tools that are then applied to 3D objects
- 3D models animated using skeletal animation that react to user input to change active animations using blending. For example, an animated character that can change from an Idle animation sequence to a Walking sequence when the user interacts with a GUI.

All source code and project files must also be submitted for assessment, including any and all assets used.

You will also be assessed on your ability to follow good coding practices throughout the development of both projects.

Present and Record Feedback

Demonstrate one, or both of your applications to end users. Record their feedback along with any adjustments you made to your implementations in response to this feedback.

Write a brief report that list the feedback, your response to the feedback, and a brief evaluation of the technologies, techniques and programs used in your implementation(s). In your evaluation, pay special attention to any practical implications the technology may have, especially in relation to the cross-discipline team projects you will be completing later in the year.

Evidence Specifications

This is the specific evidence you must prepare for and present by your assessment milestone to demonstrate you have competency in the above knowledge and skills. The evidence must conform to all the specific requirements listed in the table below. You may present additional, or other evidence of competency, but this should be as a result of individual negotiation with your teacher.

Your Roles and Responsibilities as a Candidate

- Understand and feel comfortable with the assessment process.
- Know what evidence you must provide to demonstrate competency.
- Take an active part in the assessment process.
- Collect all competency evidence for presentation when required.

This table defines what you need to produce as evidence of competency.

Assessment Tasks & Evidence Descriptions
<p>1. Completed Real-Time 3D OpenGL Application</p> <p>Evidence that includes:</p> <ul style="list-style-type: none"> • Submitted stand-alone executable for a 3D real-time application that implements OpenGL, which must run error-free and demonstrate the following in a single scene: <ul style="list-style-type: none"> ○ 3D models rendered with custom GLSL shaders ○ Texture mapping ○ 3D lighting • Submitted source code and assets for OpenGL 3D application
<p>2. Completed Real-Time 3D Application</p> <p>Evidence that includes:</p> <ul style="list-style-type: none"> • Submitted stand-alone executable for a 3D real-time application that was created within a game engine or custom 3D rendering engine that runs error-free • Application demonstrating the following features within a single scene:

<ul style="list-style-type: none"> ○ 2D GUI that can be interacted with by the user and interacts with the 3D world in some manner ○ Custom materials applied to 3D objects ○ Skeletal animation that reacts to user input ● Submitted project source and assets for in-engine 3D application
<p>3. Present and Record Feedback</p> <p>Evidence that includes:</p> <ul style="list-style-type: none"> ● Application(s) demonstrated to end users and feedback recorded ● Implementation(s) adjusted in response to feedback ● Feedback and adjustments are recorded and submitted in a document, along with an evaluation of the technologies, techniques and programs used in the implementation
<p>4. Follow Good Coding Practices</p> <p>Evidence that includes:</p> <ul style="list-style-type: none"> ● Applications debugged and tested to ensure they run error-free ● Code following consistent naming conventions ● Files commented to an acceptable industry standard as specified by your instructor

Assessment Instructions for Candidate

METHOD OF ASSESSMENT

Assessment is a cumulative process which takes place throughout a subject. A 'competent' or 'not yet competent' decision is generally made at the end of a subject. Your assessment will be conducted by an official AIE qualified assessor. This may be someone other than your teacher. The evidence you must prepare and present is described

above in this assessment criteria document. This evidence has been mapped to the units of competency listed at the beginning of this document. Assessments will be conducted on a specific milestone recorded above in this assessment guide document.

ASSESSMENT CONDITIONS

Formative assessment takes place as your teacher observes the development of your work throughout the subject and, although the assessor is likely to be aware of the evidence you are submitting, it is your responsibility to be prepared for the interview where a competency judgement is made (summative assessment). Forgetting something, or making a small mistake at the time of the milestone assessment, can be corrected. However, the assessor may choose to assess other candidates who are better prepared and return to you if time permits.

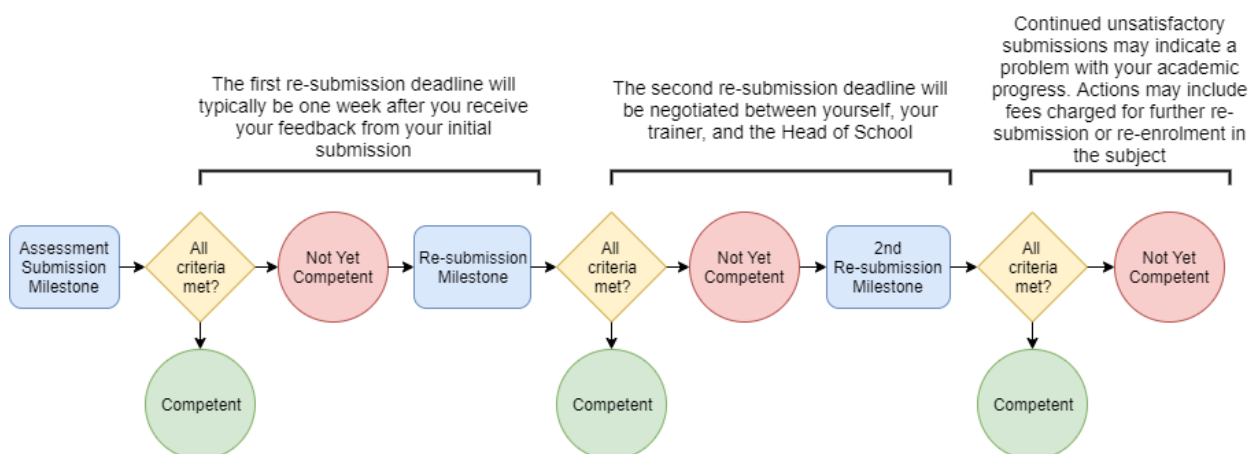
Upon completion of the assessment you will be issued with feedback and a record of the summative assessment and acknowledge that you have received the result. If you are absent for the nominated

assessment milestone (without prior agreement or a sufficiently documented reason) you will be assessed as not yet competent.

GRADING

The assessment you are undertaking will be graded as either *competent* or *not yet competent*.

REASSESSMENT PROCESS



If you are assessed as being not yet competent you will receive clear, written and oral feedback on what you will need to do to achieve competence. Failing to submit an assessment will result in you being assessed as not yet competent. You will be given a reassessment milestone no more than one (1) week later to prepare your evidence. If you are unsuccessful after your reassessment, you may be asked to attend a meeting with your Head of School to discuss your progress or any support you may need and further opportunities to gain competency.

REASONABLE ADJUSTMENTS

We recognise the need to make reasonable adjustments within our assessment and learning environments to meet your individual needs. If you need to speak confidentially to someone about your individual needs, please contact your teacher.

FURTHER INFORMATION

For further information about assessment and support at AIE, please refer to the assessment and course progress sections of your student handbook.

Software

Core

Microsoft Visual Studio

Microsoft's Visual Studio is the recommend IDE for this subject. Other IDEs may be employed if desired as the content of this subject is designed to be cross-platform and IDE agnostic though we cannot guarantee that all subject material will operate as intended on other IDEs and platforms.

- <https://www.visualstudio.com/vs/>

Unity3D / Unreal Engine 4

Projects need to be done in a game engine of choice. Learners can use the engine of their choice, no restrictions, or a custom (self-programmed) 3D rendering engine.

Unity 3D is a widely used 3D game engine. It has powered many financially and critically successful games. It has a wide array of features that aid with development, especially for a small team. Games made with Unity can be built to a large array of devices.

- <http://www.unity3d.com>

Unreal Engine 4 is a complete suite of game development tools used to make games from 2D mobile games to console blockbusters and VR. Unreal 4 is a 3rd party development tool used in many game studios and offers professional development experience.

- <https://www.unrealengine.com/>

References and additional material

Useful Web Links

- [Rendering \(computer graphics\) Wikipedia Article](#)
- [Real-Time Rendering](#)
- [Real-Time Rendering Wikipedia Article](#)
- [OpenGL Home Page](#)
- [GL Programming](#)
- [OpenGL Tutorials](#)