SPECIALIST EDUCATORS IN GAMES, ANIMATION & FILM VFX

2021 Subject & Assessment Guide

Complex Game Systems

10702NAT

Advanced Diploma of Professional Game Development

Programming

academy of interactive entertainment

Sydney Melbourne Canberra Adelaide Lafayette Seattle Online

Table of Contents

Complex Game Systems

Units of Competency

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

PGDGSP6006 – Develop complex systems for real time applications

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

Overall Learning Outcomes

- Implement networking for games and simulations
- Apply an understanding of threading and parallel programming techniques
- Use audio in game programming
- Implement automated systems for building and testing

Subject Description

Games and simulations are full of myriad algorithms and techniques. This subject combines many different topics and techniques that are needed in various areas of game programming.

The core learning focuses on the topics of networking and multithreaded programming. There are also several advanced bonus topics that may be covered during class or assigned as homework. Over the course of this subject it is expected that you will cover both core topics and at least one bonus topic.

In the topic on multithreaded programming, students will dive into multithreaded and parallel programming techniques that are essential in modern multicore environments. Computers are full of various processing units and being able to harness their power means our games can be pushed even further.

The topic on networking looks into setting up our own networked architecture to allow clients and servers to communicate with each other. The majority of games developed have some form of network communication, so we will be looking into how we can incorporate this into our own games. Topics are explored through low level network programming in C++ and through the use of networking APIs exposed in popular 3D game engines.

Additional topics include procedural content generation, exploring how components of our game worlds, such as terrain or level data, can be created programmatically through advanced algorithms; artificial intelligence on the GPU covering how advanced AI algorithms can be executed on the graphics card to enable dramatic improvements in execution time; automation for software development examining creating automated build servers to continuously build our applications and automated testing to ensure our applications run as expected; and scripting within games and how we can integrate a scripting language into our applications.

Industry Relevance

Games and simulations use various techniques that this subject explores. Networking is a key factor for many successful games and this subject teaches the fundamental knowledge needed for creating networked games.

Many simulations and new emerging software engineering fields require knowledge of machine learning systems for various areas including analytics and research. Executing sophisticated artificial intelligence algorithms on the GPU offers performance increases of orders of magnitude over traditional algorithms executed on the CPU allowing games and simulations to achieve a new level of quality and quantity.

Many games make use of scripting to allow fast iterations on development allowing for designers to implement core gameplay mechanics without the need of recompiling the game.

Procedural content generation can help reduce the cost of game development as game content is created using advanced algorithms rather than by a traditional artist or designer. Game companies, large and small, use such techniques within games to populate worlds or develop content that would be prohibitively expensive to do by hand.

And with modern computer architecture involving dozens to thousands of processing units parallel programming techniques are becoming more crucial for games and simulation needs. Being able to calculate results from a complex simulation at real-time rather than waiting hours to days has had a massive impact in advancing various fields including science and medicine.

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

Subject Textbooks

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

- Jason, G., 2018. ***Game Engine Architecture*** *Third Edition*. Taylor & Francis Group.
- B., D., 2016. ***Programming Massively Parallel Processors: A Hands-on Approach***. Morgan Kaufmann.

Assessment Criteria

Assessment Description

Assessment Milestones

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

General Description

For this assessment you are required to implement a modular complex system to be used within a test application. The modular system may be created as a statically-linked library (.lib), a dynamic-link library (.dll) or as redistributable source code that a user compiles into the project. Ffor example, a package for Unity 3D to be distributed via the asset store, a C++ header library, or some other sufficiently decoupled source code implementation.

The Modular Complex System: Brief and Implementation

The specific evidence requirements for the brief and implementation are detailed in the *Modular Complex System Brief* and *Implement Modular Complex System* items in the table *Assessment Tasks and Evidence Descriptions* below.

The complex system could be one of many different systems based around the concepts covered in the various topics for this subject, or an extension of a suitably complex topic covered in another subject.

Examples of suitable modular systems you could develop include (but are not limited to) a multi-threaded task manager, a network update downloader, a procedural dungeon generator, or a pathfinding algorithm implemented on the GPU. You are to discuss ideas with your teacher or your teacher may provide you with potential example projects.

Once you have a chosen project that your teacher approves of, you are to create a project brief detailing:

- the purpose of the system,
- any libraries it relies on,
- the mathematical operations to be used,
- the advanced algorithms to be implemented,
- how it will be made modular, and
- how to integrate your system with a new or existing application.

This brief must be approved by your teacher before you can continue to the implementation phase of the project.

Once approved, you are to build your modular system.

The Proof of Concept Application

This specific requirements for this piece of evidence is detailed in the *Integrate Modular Complex System* item in the table *Assessment Tasks and Evidence Descriptions* below.

Once your complex modular system has been built, you need to integrate your system into an application that makes use of your system and proves that it functions as intended.

The modular complex system and the test application must function error free and compile without any errors.

Performance Evaluation Documentation

Upon completing the proof of concept test application, you are to write a brief report on the performance of your complex modular system.

Include in your report the following information:

- Issues encountered when integrating the modular complex system into the test application. Be sure to outline any changes required when implementing your system that were different to the details included in your initial project brief.
- The performance of the system.
  How you measure performance will be dependent on the type of complex system implemented. However, you should aim to benchmark your system against similar implementations.
  For example, for an AI algorithm implemented on the GPU, compare and contrast against the same algorithm running on the CPU. For a networking application, measure latency and performance across a range of network conditions. For a procedural generation algorithm, an analysis of the algorithm's complexity or performance (with reference made to Big O) may be sufficient.
  Make special note of any optimizations included in your implementation or areas where improvements could be made.
- Any required changes for the system to function as intended.

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
1 Modular Complex System Brief Evidence that includes: ● Creation of a written document that details and describes a Modular Complex System that you intend to implement that must include:

○ Purpose of the system ○ Additional libraries the system will require ○ If the system is to be implemented as a redistributable static or dynamic linker library or as redistributable source code ○ The mathematical operations used ○ The advanced algorithms to be implemented ○ Information on how the system should be integrated into an application ● The system must be modular, being able to be added to various projects to serve its purpose ● Your teacher will advise you on your system and may provide you with example briefs or suggested systems to implement
2 Implement Modular Complex System Evidence that includes: ● Modular Complex System implemented as per the specified brief ● Source code and assets for the system submitted for review and assessment ○ Source code is free from compile and link errors ● If implemented as a redistributable static or dynamic linker library, the library must also be submitted
3 Integrate Modular Complex System Evidence that includes: ● An application that integrates and makes use of your Modular Complex System ● A written document detailing: ○ Issues encountered integrated the Modular Complex System ○ Performance of the system ○ Any required changes for the system to function as intended ● Application executable submitted than can run external to an IDE without errors ● Source code and assets submitted for review that contain no compile or link errors

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

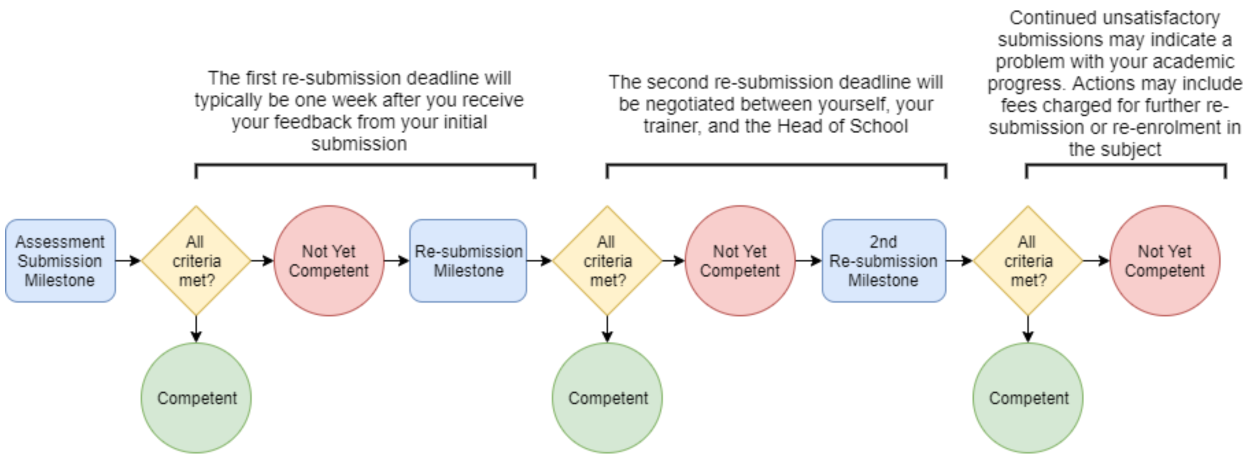
Formative assessment takes place as your teacher observes the development of your work throughout the subject and, although the assessor is likely to be aware of the evidence you are submitting, it is your responsibility to be prepared for the interview where a competency judgement is made (summative 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


The first re-submission deadline will typically be one week after you receive your feedback from your initial submission
The second re-submission deadline will be negotiated between yourself, your trainer, and the Head of School
Continued unsatisfactory submissions may indicate a problem with your academic progress. Actions may include fees charged for further re-submission or re-enrolment in the subject
Assessment Submission Milestone
All criteria met?
Not Yet Competent
Re-submission Milestone
All criteria met?
Not Yet Competent
2nd Re-submission Milestone
All criteria met?
Not Yet Competent
Competent
Competent
Competent

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 recommended 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, however we cannot guarantee that all subject material will operate as intended on other IDEs and platforms.

- https://www.visualstudio.com/

OpenCL / CUDA

There are multiple GPGPU libraries available for use to use within the course with OpenCL being available on most platforms. CUDA was developed by Nvidia and currently is only available on their platform from their website. On OSX OpenCL is already included, but for Windows and Linux platforms you will need to obtain the library from your GPU's hardware vendor, such as AMD, Nvidia or Intel.

- https://developer.nvidia.com/cuda-downloads
- http://developer.amd.com/tools-and-sdks/opencl-zone/
- https://software.intel.com/en-us/intel-opencl

RakNet

RakNet is an open-source networking library primarily targeting games. It is available from Oculus on their github page.

- https://github.com/OculusVR/RakNet

FMOD / Wwise

FMOD is an Australian-developed audio SDK for games and real-time applications that allows us to easily integrate sound. Wwise is a similar library. Both are extremely popular within game development.

- http://www.fmod.org/
- https://www.audiokinetic.com/products/wwise/

Lua / Python

Many scripting languages exist that have been used within games, with Lua being the most commonly integrated language due to its small size and ease of integration. Python is another popular choice due to its robust language and support.

- http://www.lua.org/
- https://www.python.org/

Unity3D / Unreal Engine 4

Some topics will be discussed within the context of a game engine. Learners will also have the option of developing their projects either as a custom application or within a game engine of their choice, no restrictions.

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/