ANIMATION

AUDIO

DESIGN

GAD173 Game Programming

FILM

GAMES

WEB & MOBILE

Unit Details

Module Code and Name	GAD173 Game Programming
Course	Associate Degree of Games Development Bachelor of Games Development (Game Programming)
Award Level	AQF5
Module Level	Introductory
Delivery Mode	Face to Face, Blended
Duration	1 Trimester
Credit Points	10 / 240
Pre-requisites	GAD170 Game Scripting
Co-requisites	

Overview

In order to create extensible, efficient and optimised software, programmers are often required to have more intimate knowledge of how the software and hardware upon which their programs are executed.

In this module you will be introduced to the fundamentals of systems programming as well as industry-standard practices for software development. You will be building upon and extending the knowledge from previous modules to explore lower level system aspects including memory management and how computers execute programs. You will learn how to gather requirements, prepare appropriate technical design documentation and then deliver a code-based solution based on that design.

This document is part of an information pack which contains a number of resources that will guide you to success in this Module. Contained within this document;

1. Workload Expectations
2. Content Knowledge to be covered in this Module
3. Learning Outcomes
4. An overview of the milestones
5. How you will be assessed
6. How we will guide you to develop your transferable skills
7. The resources that you require to complete activities outside of class time.

Workload

Facilitated Learning	Self Directed Learning	Total workload hours per week
3	**6**	**9**

Required Content Knowledge

To achieve all of the specified learning outcomes, your Facilitator will ensure that you are adequately briefed on the following content areas;

Technical Design Documentation
- Requirements gathering
- UML diagrams
- User Interface mockups
- Testing methodologies

Systems Programming
- Introduction to systems programming
- Process of building and running an executable
- Static vs dynamic memory allocation
- Value vs reference types
- File I/O
- Design patterns
- Data structures
- Performance profiling and algorithm analysis
- Version control systems

Object Oriented Programming
- Classes
- Inheritance
- Polymorphism

Milestone Overview

Throughout the trimester, your Facilitator will evaluate your performance and activities against the requirements for this Module, which are described in detail below. Your Facilitator will guide you to develop your Transferable Skills and to achieve the Baseline Assessment requirements by the end of the module. Your Baseline Assessment, Transferable Skills performance, and quality of the work presented will make up your final grade.

Monitoring your Progress

Each week, you will be assigned a set of tasks to complete before, during and after class. These tasks will be a mixture of collaborative and independent activities done face to face and online each week. Engaging in the before, during and after class activities is important for your progress in the modules.

Before class, you Seek and understand ideas that are applied and Acted upon in class. After class, you Evaluate, practice and reflect to build on what you have learned. These activities are connected with your

projects which ultimately determine your success in the module. Your Learning Journal is the central place where your activities and the feedback you receive are noted and reflected upon.

In Stage 1 of your program, projects have defined Briefs and Milestones. The Projects are designed so that successful completion will enable you to reach Baseline and your engagement with the Brief processes will contribute to your Holistic grade.

To earn a grade higher than a pass (Baseline), you will need to go above the minimum requirements of the module. The holistic grading criteria explain what you need to do in greater detail. Although your facilitator will give you feedback on both your baseline and holistic performance, you will be expected to reflect on your own performance at specific intervals throughout the trimester. This is an important aspect of developing your creative media skills as it will give you insight into your own performance which will be a valuable skill for you as you move to become an effective creative media professional.

Baseline Assessment

There are specific Learning Outcomes which must be met for this Module. Note that all Baseline criteria must be satisfied within the trimester to pass the Module. You and your Facilitator will use this list to track your progress over the course of the trimester. You will have up until Week 13 to satisfy all of the Learning Outcomes. To receive a grade of higher than a pass, you will need to maintain your Learning Journal and reflect on your own progress throughout and at the end of the trimester.

Many of the tasks that you will undertake are matched to Learning Outcomes. The diagram below (Figure 1) illustrates how the projects you do can be linked to Learning Outcomes. Refer to the Holistic Assessment document for further information.

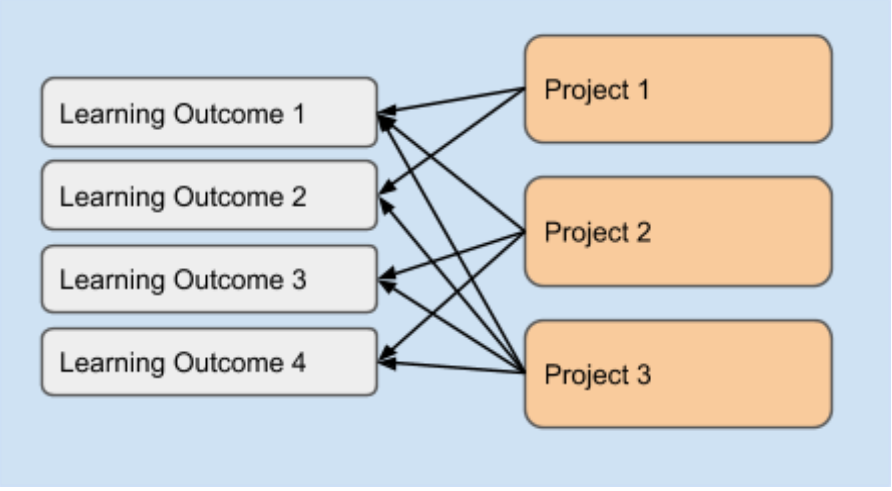

Figure 1

GAD173 Learning Outcomes

	Learning Outcome	Baseline Criteria	GAs
LO1	Gather requirements, design a code-based solution to meet those requirements and then deliver the code-based solution.	Context: Identifying and gathering the requirements on a project is critical to understanding how code-based tools can be utilised to effectively solve problems. Programming languages like C and C++ have common paradigms for use in developing code. This includes the data structures, compiling, and execution function. Understanding these is critical to your effective use of the language to build software. Range of Approaches: - Gather the requirements of the project, define and implement code based functionality using a C based (or appropriate) programming language, Variables and Types, Constants, Operators, and the Standard Library. Performance Standards: - You gather and define requirements of the project using industry standard tools such as UML and use cases. - You correctly implement the standard structure of c based language including; Variables and Types, Constants, Operators - You make effective use of Standard library. - You document the implementation of code through clear code comments.	Graduate Attributes; 1, 2, 3, 4, 5, 6.
LO2	Apply procedural and object oriented programming techniques to solve problems.	Context: Object Oriented Programming allows large, complex programs to be written. It allows large groups to work effectively. And allows problems to be solved using the natural language/terms of the problem domain. OOP language features allow the programming of an application to be accelerated and reduce the opportunity for error and duplicate code. Range of Approaches: - Implement object oriented programming language features such as classes, inheritance, and polymorphism into your game project. Performance Standards: - You correctly describe the use of Object Oriented Programming language features. - You have implemented classes, polymorphism, and inheritance as appropriate in the project. - You have documented the implementation through clear code comments.	Graduate Attributes; 1, 2, 3, 4, 6.
LO3	Critically evaluate your own development processes, technical designs and technical implementations.	Context: Successful practitioners reflect on the successes and failure of their craft. The questioning and reflecting on the processes we use to produce our creative work helps us improve as creative practitioners and use of technical tools such a programming languages, features, and algorithms. Range of Approaches: - Written postmortem of completed work - Unit test/TDD/etc. of your ongoing work - Peer review Performance Standards: - You have critically reflected on your technical design and code identifying specific attributes of this. - You have undertaken a process of presenting this information to others and seeking peer feedback. - You have articulated processes or practises to improve the technical implementation. - You propose a range of potential alternative solutions to identified problems - You correctly identify design patterns structural elements of your technical designs	Graduate Attributes; 1, 2, 4, 5, 6, 7.
LO4	Apply appropriate industry-standard practices to the design, development and delivery of code-based solutions.	Context Games are a team activity, with many interconnected and interdependent moving parts. As such programmers are required to actively assist in planning and scheduling of work to be done on the project. Range of Approaches: - Select appropriate tools - Select IDE - Identify relevant coding standards	Graduate Attributes; 1, 2, 5, 6, 7.

		- Propose workflows to ensure quality and functionality - Identify contemporary software development trends and propose selection for a project of small scale - Create a tool for others - Implement underlying game engine systems - Create game engine - Extend and existing tool or engine Performance Standards: - You maintain a work or task backlog - You make effective use of source control - You ensure coding standards are adhered to during development - You have identified and prioritised risks within the project - You have undertaken a testing and validation process to ensure functionality is correct. - You have managed work within milestones and deliverables - You make use of contemporary software development methods	

Assessment Overview:

Your results for this module, GAD173 Game Programming, are based on the following elements.

Assessment	Assessment Description
Baseline Learning Outcomes	All Baseline Learning Outcomes must be completed to pass this module.
Projects	Projects are pre-defined activities with specific Briefs, Milestones, and standards. They are mapped directly to the Baseline Learning Outcomes and are provided as separate documents via Campus Online
Transferable Skills	Transferable Skills Framework must be addressed to pass this module. Your Learning Journal reflections and your engagement in class will indicate your development of transferable skills
Final Reflection	A Final Reflection is required to pass this module. Throughout the trimester, each of you will present your work to your peers and facilitator(s) as part of the project work and Learning Journal. The Final Reflection is a summary of your folio of work and reflections on your learning. Use this folio presentation to reflect on these aspects: 1. What you have created in the trimester 2. What you have learned in the trimester

	3. What you have learned about your own practice during the trimester 4. What you plan to do to improve next trimester 5. Where you think you fit on the overall grade scale (see the Overall Assessment Guide for details). The format of the reflection may be a page of your Learning Journal, a linked video or other forms as defined by a separate brief.
Holistic Assessment	Once Baseline has been achieved, a holistic grade is assigned between Pass and High Distinction. See the Holistic Assessment Guide for details.

Transferable Skills

Working in any professional environment requires strong transferable skills. Often these skills are more difficult to develop than technical skills. During your work in Modules, you have the following Transferable Skills (TSs) that you will be expected to work towards;

1. Self Reflection
2. Deliberate practice
3. Time Management Abilities
4. Strong Work Ethic
5. Work Well Under Pressure
6. Communication Skills
7. Act as a Team Player/ Ability to collaborate
8. Problem-Solving Skills
9. Critical Thinking
10. Cognitive Outsourcing
11. Self-Confidence
12. Positive Attitude
13. Ability to Accept and Learn From Criticism
14. Flexibility/Adaptability

Prescribed Reading(s):

Alexandrescu, A. (2001). *Modern c design : Generic programming and design patterns applied*. Boston, MA: Addison-Wesley.
Knuth, D. (2011). *The Art of Computer programming, Volumes 1-4a*. Boston, MA: Addison-Wesley.
Larman, C. (2005). *Applying uml and patterns : An introduction to object-oriented analysis and design and iterative development* (3rd ed.). Upper Saddle River, NJ: Prentice Hall.

Recommended Reading(s):

Backhouse, R. (2011). *Algorithmic problem solving*. Chichester, England: Wiley Publishing.
C++ FAQ (n.d). Retrieved from https://isocpp.org/faq
Kreinin, Y. (2007). *C++ FQA Lite: Main page*. Retrieved from http://yosefk.com/c++fqa/index.html
C++ language reference (n.d). Retrieved from http://msdn.microsoft.com/en-us/library/3bstk3k5.aspx
C++ Programming (n.d.). Retrieved from http://en.wikibooks.org/wiki/C%2B%2B_Programming
C++ reference (n.d.). Retrieved from http://en.cppreference.com/w/cpp
Cplusplus.com/ (n.d.). Retrieved from http://www.cplusplus.com/

Gamma, E. (1995). *Design patterns : Elements of reusable object-oriented software*. Reading, MA: Addison-Wesley.
Larman, C. (2005). *Applying uml and patterns : An introduction to object-oriented analysis and design and iterative development* (3rd ed.). Upper Saddle River, NJ: Prentice Hall.
Learn Visual c++ (n.d).Retrieved from http://msdn.microsoft.com/en-us/vstudio/hh386302
Learncpp.com (n.d). Retrieved from http://www.learncpp.com/
Liberty, J., & Jones, B. (2005). *Sams teach yourself c in 21 days* (5th ed.). Indianapolis, IN: Sams.
Meyers, S. (2001). *Effective stl : 50 specific ways to improve your use of the standard template library*. Boston, MA: Addison-Wesley.
Meyers, S. (2005). Effective c : 55 specific ways to improve your programs and designs (3rd ed.). Upper Saddle River, NJ: Addison-Wesley.
Schmuller, J. (2004). *Sams teach yourself uml in 24 hours* (3rd ed.). Indianapolis, IN: Sams.
Sutter, H., & Alexandrescu, A. (2005). *C coding standards : 101 rules, guidelines, and best practices*. Boston, MA: Addison-Wesley.
eTutorials UML (n.d). Retrieved from http://etutorials.org/Programming/UML/
Roam, D. (2009). *The back of the napkin : Solving problems and selling ideas with pictures*. London, England: Marshall Cavendish.
McConnell, S. (2004). *Code complete* (Second ed.). Redmond, WA: Microsoft Press.
Tremblay, C. (2004). *Mathematics for game developers*. Boston, MA: Thomson Course Technology.
Van Verth, J.M., Bishop, L.M. (2008). *Essential mathematics for games and interactive applications: A programmer's guide* (2nd ed.). Burlington, MA: Morgan Kaufmann.

Resources and Facilities

Word Processor
Subject specified Integrated Development Environment w/ visual c++
Subject specified UML creation software
Text editor with code hinting

Group Work

Your information pack contains a document called **Group Work Policy**. This outlines the individual team roles, responsibilities and risk mitigation strategies for students working in a group. It is essential that all group members read and understand this before undertaking group work. These guidelines look at four key responsibilities for team members:

- Equity
- Communication
- Deliverables
- Learning

Appendix: Supporting Documents

Student Handbook -
https://drive.google.com/open?id=1fCl4HyBvMW-Opl02YsRKuTogBdbgeRxl

Holistic Assessment -
https://drive.google.com/open?id=1Qfyl8hR1zpmfiia3fdYFsdbrFVnbaBeI

Transferable Skills -
https://drive.google.com/open?id=1hFbVCq6U3Inoj7iLFbc1xW9erG8Gn-9C

Group Work Policy -
https://drive.google.com/open?id=1r-2h81Cp4f0mj5Ig4pALHTybwz_bpwj_

Log of Changes to Module Guide

Version Number and Trimester of Introduction	Updates to LOs	Updates to Assessment	Updates to Content
101 - 19T2	Document created	Document created	Document created