

2022 Subject & Assessment Guide

Maths for Games

ICT50220 Diploma of Information Technology
(Game Programming)

CUA51020 Diploma of Screen and Media

Table of Contents

Maths for Games	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

Maths for Games

Units of Competency

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

PGDMTH6005 – Apply fundamental games programming mathematics skills

[CUADIG511](#) – Coordinate testing of interactive media products

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

- Apply fundamental mathematical skills needed for games and simulation programming
- Apply a Knowledge of Vector and Matrix maths for 2-dimensional and 3-dimensional applications
- Implement basic collision detection
- Perform basic binary operations

Subject Description

Game and simulation programming is heavily reliant on mathematical techniques to manipulate and display their virtual environments.

This subject aims to teach you the fundamental mathematical skills needed in 2-dimensional and 3-dimensional games, starting out with the basic mathematical formulae and then moving on to more complex techniques involving Vectors and Matrices.

Throughout this subject you will work through implementing foundational maths concepts in code in order to fully understanding the formulae needed to manipulate the Vectors and Matrices used within all games. To ensure the mathematical formulae are correct, the data types you write will be tested with a Unit Testing application to ensure that the output is correct.

You will also explore basic collision detection techniques used in video games to determine when objects come into contact with each other as well as gain an understanding of binary operations within programming.

Industry Relevance

Mathematics is essential in all video game and simulation work, from basic mathematical algorithms and formulae right up to complex calculations and simulations needed for advanced techniques such as physics and film lighting.

Even when using modern game engines contain their own vector and matrix implementations, a solid grounding in the application of mathematic concepts is essential for developing solutions to common problems found in video games or implementing game mechanics.

Assumed Knowledge

- Knowledge of C# programming
- Introductory high school-level mathematics knowledge

Subject Textbooks

The following textbook is highly recommended for this subject:

- Dunn, P, Parberry, I, ***3D Math Primer for Graphics and Game Development***, 2nd Edition, CRC Press (2011)

The following textbooks are also recommended:

- Lengyel, E, ***Mathematics for 3D Game Programming and Computer Graphics***, 3rd Edition, Cengage Learning (2011)
- Ericson, C, ***Real Time Collision Detection***, CRC Press (2005)

Assessment Criteria

Assessment Description

Assessment Milestones

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

General Description

In this subject, you will program custom data types that implement Vector and Matrix objects used in games.

Your first task is to write a collection of custom data types in C#. These will implement the required data types (either as structures or classes) and functions listed below. To ensure that your code functions correctly, you will use a Unit Test Application to test the accuracy of your mathematical methods.

You will also create a 'technology demonstration' that highlights the usefulness of your maths data types by performing transformations within a graphical Test Application. This graphical application must use your custom data types to transform and display items and use a matrix hierarchy to display visual elements correctly. Your trainer will advise on setting up the visual components or provide you with some framework code to build on.

The graphical test application can be a small demonstration only, not a complete game. A suggested project brief is provided for you, although you can create an application of your choosing upon negotiation and approval from your trainer.

The final part of this assessment is the completion of several Binary to Decimal number conversion exercises.

The Custom Math Data Types

Your math data types (i.e., structs or classes) must include the following types. The maths operations should be in column-major order, using a right-handed coordinate system, with the following names of types and functions to ensure that your code works correctly with the Unit Test Application:

- Vector3 (3-dimensional vector using 3 floats)
- Vector4 (3-dimensional homogeneous vector using 4 floats)
- Matrix3 (2-dimensional 3x3 rotation matrix using 9 float)
- Matrix4 (3-dimensional 4x4 transform matrix using 16 float)
- Colour (RGBA values stored as a 4 byte integer)

Your types must overload the following mathematical operators and include the additionally mentioned member functions, where **V** represents a Vector, **M** represents a Matrix, **n** represents an index and **f** represents a float:

- $V = V + V$ (point translated by a vector)
- $V = V - V$ (point translated by a vector)
- $V = V \times f$ (vector scale)

- $V = f \times V$ (vector scale)
- $V = M \times V$ (vector transformation)
- $M = M \times M$ (matrix concatenation)
- $f = V.\text{Dot}(V)$
- $V = V.\text{Cross}(V)$
- $f = V.\text{Magnitude}()$
- $V.\text{Normalise}()$
- $M.\text{setRotateX}(f)$
- $M.\text{setRotateY}(f)$ and
- $M.\text{setRotateZ}(f)$

Your types should all have default constructors, and constructors that allow each float element to be set individually, for example:

- `Vector3 v3; // default`
- `Vector4 v4(0, 0, 0, 1);`
- `Matrix3 m3 = Matrix3(1, 0, 0, 0, 1, 0, 0, 0, 1);`

Your `Colour` type will have get and set functions for each colour component that employ bit shifting and bitwise operators to query or modify the individual colour components of the 4 byte integer. For example:

- `byte GetRed();`
- `void SetRed(byte value);`

Unit Tests

On *Canvas* you will find a unit test project. This project contains a series of pre-programmed tests that will validate the implementation of your custom math data types.

To demonstrate competency in this subject your custom data types must pass all unit tests provided.

Number Conversion Exercises

In addition to the math types described above, you will need to complete the number conversion exercises listed on *Canvas*.

These exercises will guide you in the creation and testing of the above-mentioned *Colour* class. You must include your answers to the Number Conversion Exercises with the rest of your assessment submission.

Graphical Test Application

You will make a graphical application to demonstrate your custom data types.

The graphical test application can be a small demonstration only, and need not be a complete game. A suggested project brief is provided for you, although you can create an application of your choosing upon negotiation and approval from your trainer.

Your test application must include the following features:

- An example of a matrix hierarchy to manipulate visible elements

- An example of game objects moving using velocity and acceleration with vectors
- An example of simple collision detection
This may be simple collision detection (for example, bullets being destroyed when they hit the edge of the screen). Collision resolution (e.g., bouncing) is beyond the scope of this assessment.

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 Redistributable Maths Data Types</p> <p>Evidence that includes:</p> <ul style="list-style-type: none"> • Submitted redistributable structs or classes that implement: <ul style="list-style-type: none"> ○ Vector types for 3D vectors, including homogeneous 3D vectors <ul style="list-style-type: none"> ▪ Types implement methods for, in all instances, translation, scale, magnitude, normalisation, cross product and dot product ○ Matrix types for 3D matrices, including homogeneous 4D matrices <ul style="list-style-type: none"> ▪ Types implement methods for multiplication, vectors transformation, and transpose ▪ Types implement methods for setting up as rotation matrices ○ As part of the Colour type, functions for manipulating a bitfield implemented using bit shift operations • Code suitably commented to an industry-standard as specified by your teacher
<p>2. Unit Test Results and Number Conversion Exercises</p> <p>Evidence that includes:</p> <ul style="list-style-type: none"> • Maths types successfully pass the Unit Test provided to you by your instructor • Completed Number Conversion Exercises, including at least one additional Unit Test for the Colour type
<p>3. Test Application</p> <p>Evidence that includes:</p>

- Submitted executable for Graphical Test Application that makes use of your maths types to implement the following:
 - Example of matrix hierarchy to manipulate visible elements
 - Example of game objects moving using velocity and acceleration with vectors
 - Example of simple collision detection

4. Application Handover

Evidence that includes:

- A Visual Studio solution and project that compiles without errors
 - All temporary and built executable files in the obj and bin folder have been removed
- A "readme" or client document explaining how to compile, run and operate the program
- All submitted material archived in a single compressed file (zip, rar, or 7z)

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

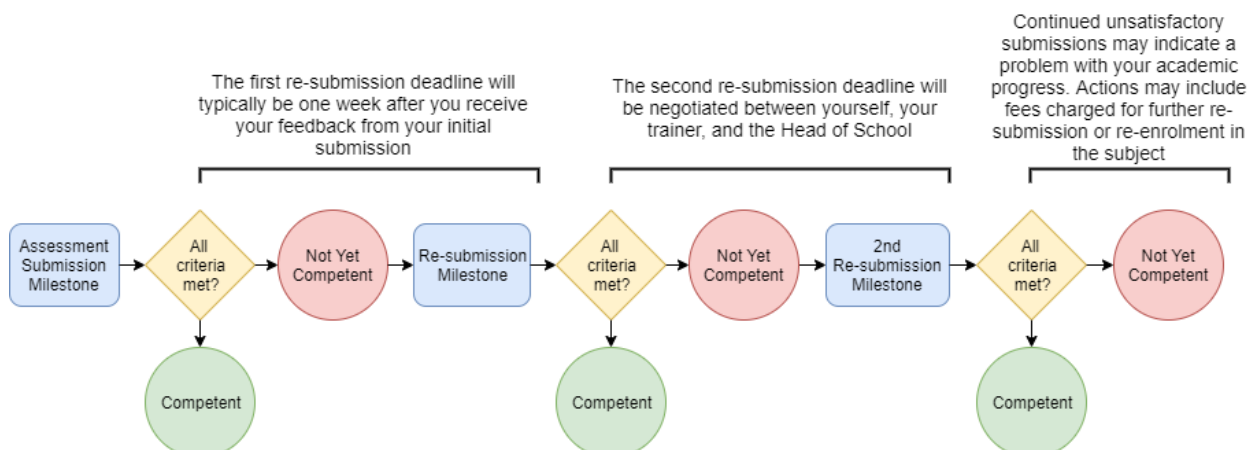
Formative assessment takes place as your teacher observes the development of your work throughout the subject and, although the assessor is likely to be aware of the evidence you are submitting, it is your responsibility to be prepared for the interview where a competency judgement is made (summative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

GitKraken

GitKraken is a leading Git GUI client for Windows, Mac and Linux, used to create and maintain version control repositories. It helps developers become more productive with Git, and provides an integrated conflict editor, built-in code editor and task tracking. Other version control clients may be employed if desired.

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

7zip

7-Zip is a free and open-source file archiver, a utility used to place groups of files within compressed containers known as "archives". This utility program will be necessary to package your assessment files for submission.

- <https://www.7-zip.org/download.html>

Microsoft Word

Microsoft Word is industry standard word processing software, development by Microsoft and used throughout the course for creating documents and reports. Microsoft Word allows documents to be saved in *word* format, as well as several other standard document formats including *pdf*.

Learners will have access to Microsoft Word on campus but may also use alternate word processing software capable of loading and saving documents in *word* or *pdf* format.

- <https://www.microsoft.com/en-us/education/products/office/default.aspx>
- <https://www.openoffice.org/>
- <https://www.google.com.au/docs/about/>