

2022 Subject & Assessment Guide

Artificial Intelligence for Games

ICT50220 Diploma of Information Technology
(Game Programming)

CUA51020 Diploma of Screen and Media

Table of Contents

Artificial Intelligence for Games.....	3
Units of Competency	3
Overall Learning Outcomes.....	3
Subject Description	3
Industry Relevance.....	3
Assumed Knowledge.....	4
Subject Textbooks	4
Assessment Criteria	4
Assessment Description	4
Assessment Instructions for Candidate	7
Software.....	9
Core.....	9
Suggested.....	9
References and additional material.....	9

Artificial Intelligence for Games

Units of Competency

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

[ICTGAM423](#) – Apply artificial intelligence in game development

[ICTICT525](#) – Identify and manage the implementation of industry specific technologies

[CUADIG517](#) – Design digital simulations

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

- Develop and apply an understanding of modern A.I. techniques for games
- Implement pathfinding algorithms
- Implement decision making techniques

Subject Description

Artificial Intelligence (A.I.) in games is a broad topic. A.I. covers a range of techniques that allow the computer to think and act based on a set of rules. The aim of most A.I. is to succeed at whatever its goal is. In the case of game A.I. its goal is actually to lose, but to lose convincingly and encourage the player to overcome a challenge.

Brian Schwab, author of “**AI Game Engine Programming**”, says the aim of game A.I. is “to be a good dad”. Imagine an A.I. entity playing a game with their kid. A bad dad would defeat their kid as quick as possible. A good dad would encourage their kid to rise to the challenge and overcome it, adapting the difficulty to meet the skills of their child.

This subject will teach you various decision-making algorithms and techniques used in games, along with locomotion techniques to move entities around your game worlds. You will also learn pathfinding algorithms that help A.I. find their way to a goal, be that a location within the level or a certain goal state such as world domination!

Industry Relevance

Artificial intelligence is a big part of modern video games that need competitive opponents and various non-player characters. Games are full of little smart agents making all kinds of decisions, from humanoids to little rodents that need to scamper around environments realistically. Simulation industries also require A.I. as does film. The Lord of the Rings movie franchise made heavy use of A.I. techniques for their hordes of orcs and humans battling it out on the big screen.

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:

- Funge, J, Millington, I, **Artificial Intelligence for Games**, 2nd Edition, CRC Press (2009)
- Mark, D, **Behavioral Mathematics for Game AI**, Charles River Media (2009)
- Buckland, M, **Programming Game AI by Example**, Wordware Publishing Inc (2005)
- Schwab, B, **AI Game Engine Programming**, 2nd Edition, Cengage Learning (2008)

Assessment Criteria

Assessment Description

Assessment Milestones

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

General Description

The assessment for this subject involves several tasks, outlined below.

The assessment tasks have been scaffolded such that each task will build on the one before it, so that by the end of this subject you will have a game or application several A.I. algorithms and systems.

Before completing each task, consider how each deliverable might be integrated into a larger game. While you do not need to build a game for this assessment, doing so will help you add context to your A.I algorithms and demonstrate their completeness.

You may wish to submit each assessment task as a separate project or combine two or more tasks into the same project. Whichever way you decide to approach these tasks, you must ensure that each deliverable is clearly identifiable within your submission.

It is recommended that your solution(s) are developed in C++ using the RayLib or AIEBootstrap framework, or another C++ graphics framework. While you are encouraged to write or use your own maths library, use of an alternative maths library (such as glm) is acceptable.

Use of alternative programming languages or full-featured game engines is strongly discouraged as it will generally be much more difficult to prove competency in writing the required algorithms. All classes and tutorials for this subject will be taught in C++ using the RayLib or AIEBootstrap framework.

Task 1: Implement the A* Pathfinding Algorithm

The tutorials for this subject discuss implementing the Dijkstra's shortest path algorithm. You must take this implementation and extend it to create a working implementation of the A* pathfinding algorithm.

Task 2: Implement one other pathfinding algorithm

If you've completed the Dijkstra's algorithm tutorial, you've already met this requirement. For those of you who are adventurous or looking to showcase your skills to future employers, you may wish to explore flow fields or navigation meshes. You could also consider polishing your implementation by integrating it into a game or similar project.

Task 3: Design and Implement a State Machine for a Non-Player Character

This is where we start crafting an NPC for inclusion in a game or simulation.

While you can follow the tutorials to add a state machine to your Dijkstra's implementation, the A.I. logic you create should be modelled on a real-world system. For this reason, the suggested approach is to design your state machine and NPC logic within the context of an application or game.

For this task, you must meet the following requirements:

- Write a game, simulation, or similar containing an agent or non-player character implementing an A.I strategy using a State Machine
- Write a detailed (1 to 2 page) design of the algorithm, including:
 - A description of the real-world environment simulated, including any technical parameters
 - A description of the underlying functionality of the A.I strategy specifying essential settings, states, conditions, and parameters
 - How the agent interacts with the simulated environment
 - Difficulty levels and their controls, as appropriate
 (Diagrams or pseudo code may also be useful to include in your design)
- Evidence that you have discussed and reviewed the design of your implementation against the project requirements

Task 4: Design and Implement One Other A.I Strategy for a Non-Player Character

Implement one other decision-making algorithm within your application (for example, decision trees, behaviour trees, blackboards, etc.). Alternatively, you may wish to showcase this A.I. strategy in a separate application or game.

As with task 3, you will document the design of your strategy.

The specific requirements for this task are:

- A game containing an agent or non-player character implementing an A.I. strategy using any other A.I algorithm
- A detailed design of the algorithm, including:
 - A description of the real-world environment simulated, including any technical parameters
 - A description of the underlying functionality of the A.I strategy specifying essential settings, states, conditions, and parameters

- How the agent interacts with the simulated environment
- Difficulty levels and their controls, as appropriate
- Evidence of discussion or review of the design against project requirements

Task 5. Write an A.I. Game Strategy Report

For this final task, write 500 words (or more) on the design and use of an A.I game strategy (such as an A.I. planner algorithm).

You do not need to write about one of the strategies you implemented (but you can). In fact, it is recommended you discuss one of the algorithms or strategies you did not implement.

You are not required to implement the strategy. This is a report only.

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 the individual requirements for each part of the assessment criteria. Listed here are the cumulative requirements for all assessment items. The evidence requirements for specific assessment items can be seen by referring to the table listed for that assessment item in the following sections.

Assessment and Competency Requirements
1. Implement the A* Pathfinding Algorithm Evidence that includes: <ul style="list-style-type: none"> • A game containing a working implementation of the A* pathfinding algorithm.
2. Implement One Other Pathfinding Algorithm Evidence that includes: <ul style="list-style-type: none"> • A game containing a working implementation of one other pathfinding algorithm.
3. Design and Implement a State Machine for a Non-Player Character Evidence that includes: <ul style="list-style-type: none"> • A game containing an agent or non-player character implementing an A.I strategy using a State Machine • A detailed design of the algorithm, including: <ul style="list-style-type: none"> ○ A description of the real-world environment simulated, including any technical parameters ○ A description of the underlying functionality of the A.I strategy specifying essential settings, states, conditions, and parameters

<ul style="list-style-type: none"> ○ How the agent interacts with the simulated environment ○ Difficulty levels and their controls, as appropriate • Evidence of discussion or review of the design against project requirements
<p>4. Design and Implement One Other A.I Strategy for a Non-Player Character</p> <p>Evidence that includes:</p> <ul style="list-style-type: none"> • A game containing an agent or non-player character implementing an A.I. strategy using any other A.I algorithm • A detailed design of the algorithm, including: <ul style="list-style-type: none"> ○ A description of the real-world environment simulated, including any technical parameters ○ A description of the underlying functionality of the A.I strategy specifying essential settings, states, conditions, and parameters ○ How the agent interacts with the simulated environment ○ Difficulty levels and their controls, as appropriate ○ Evidence of discussion or review of the design against project requirements
<p>5. A.I. Game Strategy Report</p> <p>Evidence that includes:</p> <ul style="list-style-type: none"> • A design brief (500 words) documenting the design and use of an A.I. game strategy (such as an A.I. planner algorithm) <ul style="list-style-type: none"> ○ Implementation is not required ○ The strategy may be intended for use within your submitted project(s), or in any other game

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

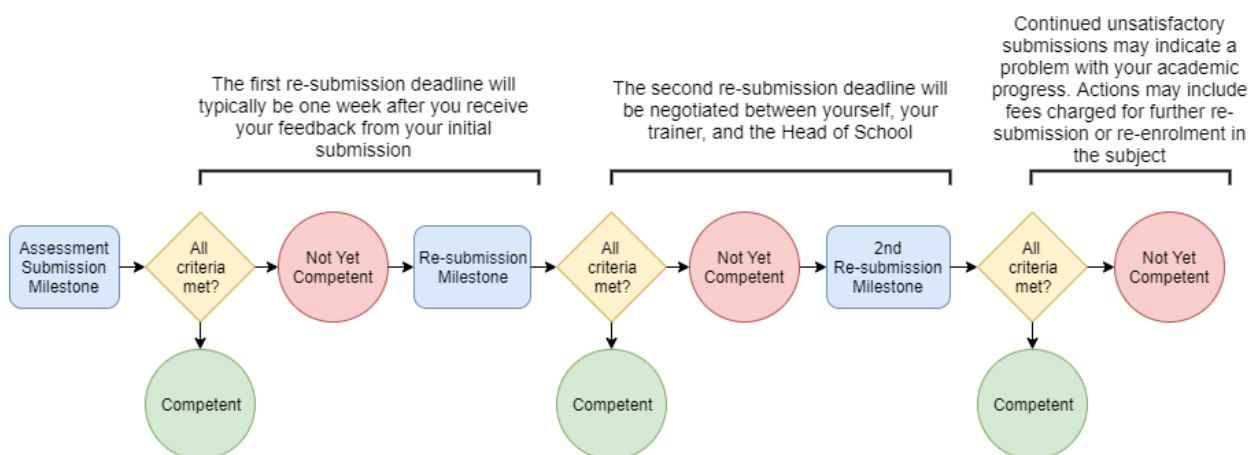
Formative assessment takes place as your teacher observes the development of your work throughout the subject and, although the assessor is likely to be aware of the evidence you are submitting, it is your responsibility to be prepared for the interview where a competency judgement is made (summative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

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/>

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>

Suggested

Photoshop / Krita

Whether it is for textures, level design or other art related topics, usually Adobe Photoshop is the software of choice for creating graphic content.

Krita is a free and open-source image editing program, offering functionality comparable to Photoshop.

- <https://www.adobe.com/products/photoshop.html>
- <https://krita.org/en/>

References and additional material

Artificial Intelligence for Games

- https://en.wikipedia.org/wiki/Artificial_intelligence_in_video_games

- <http://aigamedev.com/>
- <http://www.gameai.com/>
- [An Intro to Video Game AI for Beginners and Young Programmers \(youtube\)](#)