COS30002 Artificial Intelligence for Games

Semester 1, 2025
Learning Summary Report

Marco Giacoppo (104071453)

Declaration

I declare that this portfolio is my individual work. I have not copied from any other student's work or from any other source except where due acknowledgment is made explicitly in the text, nor has any part of this submission been written for me by another person or software service.

Signature: Marco Giacoppo

Self-Assessment Details

The following checklists provide an overview of my self-assessment for this unit.

	Pass (P)	Credit (C)	Distinction (D)	High Distinction (Low HD) | (High HD)	
Self-Assessment (please tick)		✓			

Self-assessment Statement

	Included? (tick)
Learning Summary Report	✓
Complete Pass ("core") task work, approved in Canvas	✓

Minimum Pass Checklist

Introduction

This report summarises what I learnt in COS30002 AI for games. It includes a self-assessment against the criteria described in the unit outline, a justification of the pieces included, details of the coverage of the unit intended learning outcomes, and a reflection on my learning.

Overview of Pieces Included

This section outlines the pieces that I have included in my assignment…

- **Assignment 1 Task 01: Lab - Bitbucket Setup**
  Set up source control using Bitbucket, essential for managing code versions.
- **Assignment 1 Task 02: Lab - FSM & Python**
  Implemented basic Finite State Machines (FSM) to understand AI state control.
- **Assignment 2 Task 1: Lab - Tic-Tac-Toe**
  Developed a simple AI using basic decision-making strategies.
- **Assignment 2 Task 2: Spike - Graphs, Search & Rules**
  Explored graphs and implemented early search algorithms like DFS and BFS.
- **Assignment 3 Task 1: Lab - Graphs, Paths & Search**
  Applied advanced graph-based pathfinding algorithms (including A* and GBFS).
- **Assignment 3 Task 2: Spike - Navigation with Graphs**
  Implemented pathfinding in dynamic environments using graphs.
- **Assignment 4 Task 1: Lab - Goal Oriented Behaviour & SGI**
  Created agents using Simple Goal Insistence models.
- **Assignment 4 Task 2: Spike - Goal-Oriented Action Planning (GOAP)**
  Developed agents capable of planning optimal action sequences to meet goals.
- **Assignment 5 Task 1: Lab - PlanetWars**
  Built tactical AI agents for a real-time strategy game environment.
- **Assignment 5 Task 2: Spike - Tactical Analysis with PlanetWars**
  Combined multiple AI techniques to analyze and improve tactical performance.
- **Portfolio 1 Task 1: Lab - Steering 1- Seek, Arrive, Flee**
  Basic model of steering movement for autonomously moving agents.
- **Portfolio 1 Task 2: Lab - Steering 2 - Wander and Paths**
  Continuation of previous code, with random smooth wander movement.
- **Portfolio 1 Task 3: Spike - Tactical Steering (Hide!)**
  An agent that is pursued by another while being able to hide behind obstacles.

- **Portfolio 2 Task 1: Spike - Emergent Group Behaviour**
  Cohesion, separation, and alignment steering forces on multiple agents.
- **Portfolio 2 Task 2: Spike - Agent Marksmanship**
  Targeting an agent with weapons and projectiles while calculating their future positions.
- **Portfolio 3 Task 1:-Spike - Soldier on Patrol**
  Layered state-machine design with high level modes.

Coverage of the Intended Learning Outcomes

This section outlines how the pieces I have included demonstrate the depth of my understanding in relation to each of the unit's intended learning outcomes.

ILO 1: Software Development for Game AI

"Discuss and implement software development techniques to support the creation of AI behaviour in games."

- **Lab 1: Bitbucket setup**
  → Demonstrated effective use of source control for collaborative and iterative development of AI systems.
- **Lab 2: FSM & Python**
  → Gained foundational skills in implementing state-driven AI behaviour using finite state machines.
- **Lab 3: Tic-Tac-Toe**
  → Built a rule-based decision-making agent capable of basic strategy and evaluating game outcomes.

ILO 2: Graphs and Path Planning

"Understand and utilise a variety of graph and path planning techniques."

- **Lab 4: Graphs, Search & Rules**
  → Implemented core algorithms such as DFS and BFS on abstract graph structures.
- **Lab 5: Graphs, Path & Search**
  → Applied A*, GBFS, and UCS in navigational contexts to find optimal paths between tiles.
- **Lab 6: Navigation with Graphs**
  → Used spatial graphs in dynamic environments, integrating terrain costs and obstacles for realistic agent navigation.

ILO 3: Force-based Agent Movement

"Create realistic movement for agents using steering force models."

- **Lab 11: Seek, Arrive, Flee**
  → Simulated natural motion behaviours using basic force-driven models.
- **Lab 12: Wander and Paths**
  → Introduced randomness and smooth transitions, mimicking autonomous exploration.
- **Lab 13: Tactical Steering (Hide!)**
  → Designed agents that dynamically position themselves behind cover when pursued.
- **Lab 14: Emergent Group Behaviour**
  → Created group dynamics with cohesion, separation, and alignment to simulate flocking/swarm behaviours.
- **Lab 15: Agent Marksmanship**
  → Used predictive aiming and projectile physics for AI-controlled shooting accuracy.

- **Lab 16: Soldier on Patrol**
  → Implemented a layered FSM to handle patrol and attack states fluidly, driven by triggers and context.

ILO 4: Goals and Planning Actions

"Create agents that are capable of planning actions in order to achieve goals."

- **Lab 7: Goal-Oriented Behaviour (SGI)**
  → Introduced goal insistence as a way to prioritize agent motivations.
- **Lab 8: GOAP (Goal-Oriented Action Planning)**
  → Developed agents capable of sequencing actions like cooking, gathering, and drinking based on internal needs.
- **Lab 9: PlanetWars**
  → Designed agents with tactical planning to optimize attacks and defenses using goal-based strategies.
- **Lab 16: Soldier on Patrol**
  → Extended goal-based behaviour by integrating combat tactics and environmental awareness.

ILO 5: Combine AI Techniques

"Combine AI techniques to create more advanced game AI."

- **Lab 15: Agent Marksmanship**
  → Merged predictive shooting with steering to track and hit moving targets.
- **Lab 16: Soldier on Patrol**
  → Brought together FSMs, pathfinding, and combat goals to build multi-layered, responsive AI agents.

Reflection

The most important things I leant:

Throughout the semester, I learned how to develop AI agents from scratch, starting with basic finite state machines (FSM) and building up to goal-planning and tactical AI. Understanding how each layer of AI (planning, movement, decision-making) fits together was a major learning milestone for me.

The things that helped me most were:

The labs and spike tasks were extremely helpful. All the sample codes and comments helped a lot and just made the concepts much clearer compared to just reading theories. Joining the discord server was a big help as well.

I found the following topics particularly challenging:

At first, I found combining multiple AI techniques together quite challenging. It was hard to get agents to both plans strategically and move efficiently without bugs or conflicts in behaviour. Balancing this took time.

I found the following topics particularly interesting:

I found PlanetWars lab particularly interesting. Building agents that could think ahead and adapt strategies based on dynamic conditions was very challenging but rewarding in the end.

I feel I learnt these topics, concepts, and/or tools really well:

Working on PlanetWars was a key highlight for me. It was my first time building a tactical AI system purely using Python. I learned a lot about managing complex agent behaviours and building strategic decision-making from scratch.

I still need to work on the following areas:

I need to improve how I combine multiple AI systems more seamlessly, especially for real-time games. In the PlanetWars project, I sometimes struggled to make my agents flexible enough without overcomplicating the code.

My progress in this unit was …:

I feel that my progress was steady. I consistently submitted lab work and engaged with the spike tasks, although sometimes I needed extra time to fully understand the more complex tasks. Overall, my understanding deepened over the semester, and I'm happy with my development.

This unit will help me in the future:

The techniques I learned in this unit will be useful for future game development or AI-heavy projects. Understanding how to design all of this will help me in my future plans for developments.

If I did this unit again I would do the following things differently:

I would start the spike tasks earlier and spend more time experimenting with optional challenges.

Conclusion

In summary, I believe that I have clearly demonstrate that my submission is sufficient to be awarded a credit grade.