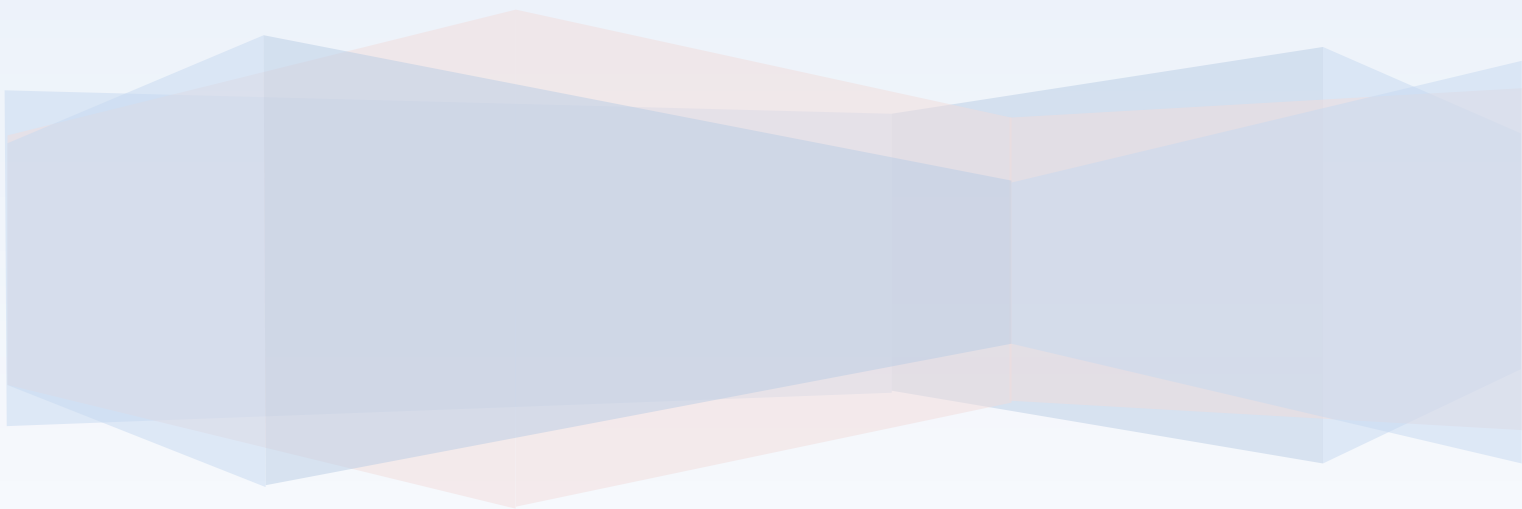


COS30002 Artificial Intelligence for Games

Semester 1, 2023

Learning Summary Report

Khanh Toan Nguyen - 104180605



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: Toan Nguyen

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

	Included? (tick)
Additional non-core task work (or equivalent) in a private repository and accessible to staff account.	✓
Spike Extension Report (for spike extensions) in Canvas	✓
Custom Project plan (for D and/or low HD), and/or High HD Research Plan document in Canvas (optional)	✓

Credit Checklist, in addition to Pass Checklist

	Included? (tick)
Custom Project Distinction Plan document, approved in Canvas	✓
All associated work (code, data etc.) available to staff (private repository), for non-trivial custom program(s) of own design	✓
Custom Project "D" level documents in Canvas, to document the program(s) (structure chart etc) including links to repository areas	✓

Distinction Checklist, in addition to Credit Checklist

	Included? (tick)
Custom Project "HD" level documents in Canvas, to document the program(s) (structure chart etc) including links to repository areas	✓

Low High Distinction Checklist, in addition to Distinction Checklist

	Included? (tick)
High Distinction Plan document, approved in Canvas	Only submitted Plan
High Distinction Report document, in Canvas, which includes links to repository assets	✓
All associated work (code, data etc.) available to staff (private repository) for your research work	✓

High High Distinction (Research) Checklist, in addition to D/Low HD Checklist

Introduction

This report illustrates as a summary of what I've learned in my course, COS30002 AI for Games. It conveys a thorough self-assessment against the criteria outlined in the unit syllabus, providing insights into my growth and development throughout the course. Moreover, the report offers a rationale behind the specific components, which shaping my understanding of AI in gaming contexts. Lastly, a reflective exploration of my learning experiences within the realm of AI for games adds depth and introspection to this narrative, encapsulating the essence of my educational journey in this dynamic field.

Overview of Pieces Included

- Github Setup: Lab 1
- FSM planning technique:
 - o Lab 2: This lab provide a brief understanding and imagination on the simple AI technique which I will learn deeper in the later of the semester.
- Tic Tac Toe AI agents:
 - o Lab 3: This lab is a implementation of simple AI agent for Tic Tac Toe game, and interaction of user to the program to choose what AI agent to play with
 - o Spike 4: Included to highlight the implementation of advanced AI techniques, specifically Alpha-Beta Pruning, in optimizing decision-making processes within the game environment.
- Navigation with Graphs:
 - o Lab 5: This lab is for getting familiar and understanding the technique behind each graph planning algorithms.
 - o Spike 6: Included to illustrate the creation of agents with different behaviors using graph-based navigation, emphasizing understanding and utilization of path planning techniques.
- Action Planning Technique:
 - o Lab 7: This lab show the simple goal oriented behaviour system (GOB) that can perform action based on selected goal, and limitation of simple goal insistence (SGI)
 - o Spike 8: Included to demonstrate the implementation of GOAP in achieving long-term strategic goals within game scenarios, showcasing proficiency in planning actions to achieve objectives.
- Tactical Analysis for PlanetWars:
 - o Lab 9: This lab used to become familiar with PlanetWars and small interaction to the program using simple Bot AI Agent.
 - o Spike 10: Included to showcase the development of a bot with strategic decision-making capabilities in a dynamic game environment, highlighting the combination of multiple AI techniques to create more advanced game AI.
- Tactical Steering:
 - o Lab 11 + 12: This lab used to get familiar with steering movement for autonomously agent, and can be able to have intergration with the agents, game environment.
- Tactical Analysis for Steering:
 - o Spike 13: Included to the implementation to create an intelligent behavior agent with complex behaviors, such as hiding and fleeing, within AI agents to enhance realism and complexity in game interactions.

- Spike 14: Included to explore the implementation of cohesion, separation, and alignment steering forces, which are fundamental for creating natural and interesting group behaviors in AI agents.
- Spike 15: Included to showcase the implementation of realistic movement and physics-based target calculations for AI agents, with a range of weapon which taking into account projectile characteristics such as speed, accuracy.
- Spike 16: Included to demonstrate the use of layered state-machine designs for agents, where high-level modes of behavior and lower-level step behaviors work together, enabling complex and dynamic agent activities.

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

The software development techniques I have mentioned in my Credit Extension Report to cover this ILO.

In addition, prior to my individual work on a custom project involving graph navigation algorithms, I created a program with an OOP structure, where each individual component is a class object to enhance software performance. Specifically, I designed an abstract class called Problem, which contains the criteria for detailed classes such as GridProblem and PuzzleProblem. This structure advances the variation of problems that can be applied to the algorithms, demonstrating my ability to design and implement flexible and scalable AI solutions in games.

ILO 2: Graphs and Path Planning

My understand of Graphs and Path Planning is demonstrate in the Credit Extension Report that covering this ILO

In addition, from my understanding of Graph and Path planning of this unit, in my custom project, I have ability to be able to implement the algorithm that is not cover from this unit and combine its technique with what I have learnt, which is bi-directional algorithms combine with BFS and Astar algorithms.

ILO 3: Force-based Agent Movement

Briefly said in here, I can be able to use physics and math to calculate and implement into the movement behavior of steering force model and the inaccuracy rate of projectile.

To demonstrate this ILO, the explanation of this I would refer to the Credit Extension task or Spike 15.

ILO 4: Goals and Planning Actions

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

The ILO for Goals and Planning Action is covered in the Credit Extension report.

In addition to that, my research for High Distinction is about AI planning technique, in more details, it is comparison of differences of AI planning technique such as Stanford Research Institute Problem Solver (STRIPS), Action Description Language (ADL) and Partial-Order Planning (POP),...

ILO 5: Combine AI Techniques

To demonstrate this ILO of AI techniques combination, Credit Extension report has briefly said on this.

Reflection

The most important things I learnt:

Throughout this unit, I had the chance to work on a variety of tasks that demonstrated autonomous agents capable of taking on tasks independently. Initially, I expected this unit to cover advanced algorithms, machine learning, or the latest technologies, given its title containing the word "AI". However, during this unit, the things that I learnt about AI in this unit is how to implement agents with behaviors that interact with their environment realistically. This doesn't necessarily mean real-life environments but rather simulations, with games serving as an excellent medium to demonstrate this concept. AI cannot be applied to real-life situations immediately without thorough testing. It must be tested in a simulated environment multiple times to ensure it operates smoothly and correctly. By taking small steps in learning how to create and implement AI in simulation games, starting with tasks like "Steering Movement," I realized that many aspects, if not thoroughly tested, can lead to numerous errors. If these errors were to occur in real-life applications, they could have significant consequences. This understanding highlighted the importance of not only programming skills but also knowledge in other scientific areas, such as math and physics, to create effective and realistic AI behaviors. Therefore, the most important things that I learnt in this unit is the ability to create and implement AI techniques into a stimulation/problem.

The things that helped me most were:

Using Canvas's unit module with lecture and YouTube was played a significant role for me to complete task. Canvas had all the important stuff like notes and slides, while YouTube had videos that showed how things work in action. By visualise things YouTube videos helped me understand better because I could see how things actually behave.

For example, the video "[Minimax: How Computers Play Games – Spanning Tree](#)" has helped me to understand minimax algorithms and its enhancement Alpha-Beta Pruning "[Algorithms Explained – minimax and alpha-beta pruning - Sebastian Lagae](#)".

For steering task, [Vectors - Basic Introduction - The Organic Chemistry Tutor](#) held be understanding of force vector with accelation. With many other video clips and so on.

I found the following topics particularly challenging:

The task that I found particularly challenging was the Steering task. This was because it required me to apply mathematical and physics principles to programming, which was quite daunting. In this task, I needed to understand concepts like velocity, acceleration, and forces, and then translate them into code to create realistic movement for the agents in the game. This involved not only understanding the mathematical formulas behind steering behaviors but also figuring out how to implement them effectively in the game environment. Overall, the Steering task pushed me out of my comfort zone and required me to stretch my understanding of both programming and physics concepts.

I found the following topics particularly interesting:

Despite being a challenging task for me, I found the Steering task particularly interesting. This task make me interest me because it required diving deep into mathematical and physics principles to create realistic movement for agents in the game. By delving into concepts like velocity, acceleration, and forces, I gained a deeper understanding of how these principles influence the behavior of objects in motion. Additionally, learning how to translate these concepts into code allowed me to see the direct application of math and physics in game development. Overall, while the Steering task presented its challenges, it also provided an opportunity for me to explore fascinating intersections between programming, mathematics, and physics.

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

I learn graph navigation well because I have a chance to intergrate with it before this unit through an online course on Youtube [Harvard CS50's Artificial Intelligence with Python](#).

Since it is my strength during this unit, I have implemented Graph Navigation problem visualisation as my custom project for this unit.

I still need to work on the following areas:

Since AI conveys a wide range of area, however, for me, the important thing I still need to work on the following areas:

Goal Planning: Goal planning involves designing AI algorithms that enable agents to plan and execute actions to achieve specific objectives within the game environment. While I have gained some understanding of goal-oriented action planning (GOAP) through projects like Spike 8, there is room for improvement in creating more robust and efficient goal planning systems. Enhancing my skills in this area will allow me to develop AI agents that can make more strategic decisions and adapt to changing game conditions effectively.

Steering Movement: Steering movement is crucial for creating realistic behavior in AI agents, as it simulates how objects navigate and interact within the game world. While I have tackled tasks related to steering behaviors in projects like Spike 13, I recognize the need to deepen my understanding of the underlying mathematical and physics principles involved. Improving my proficiency in steering movement will enable me to develop AI agents that exhibit more fluid and natural movement patterns, enhancing the overall gaming experience.

By focusing on these areas and continuously honing my skills through practice and learning, I aim to further enhance my abilities in creating sophisticated and immersive AI behaviors for games.

My progress in this unit was ...:

My progress in this unit was quite slow at the beginning. I found myself only submitting one lab task to Canvas per week and getting marked by the tutor. However, as the unit progressed, I was able to catch up with the spike tasks. Initially, I struggled with writing correct spike reports, but after submitting a couple of spike tasks and receiving feedback from my tutor, I gained a better understanding of what was expected and how to demonstrate my work effectively.

Since then, I became more proactive to my tasks and the extensions, actively planning ahead by asking more questions and seeking guidance for the custom project planning. This approach allowed me to better manage my workload and make steady progress for the remaining of the unit.

This unit will help me in the future:



If I had the opportunity to redo this unit, I would approach it with more focus from the beginning and be more proactive in asking questions with my tutor. Instead of taking a slower start, I would strive to maintain a consistent pace and seek clarification promptly whenever I got some challenges. By staying more engaged and seeking assistance when needed, I believe I could optimize my learning experience and achieve even better results.


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


The knowledge and skills acquired in this unit will be invaluable in my future studies and career. Firstly, understanding AI for games equips me with applicable skills across various industries, including gaming, simulation, robotics, and AI development. Concepts like goal planning, pathfinding, and steering behaviors are fundamental in creating intelligent systems, which are increasingly in demand in today's technology-driven world. Additionally, the problem-solving abilities and critical thinking skills developed during this unit are transferable to numerous contexts, allow me to handle complex challenges and effective solutions.

Other...:

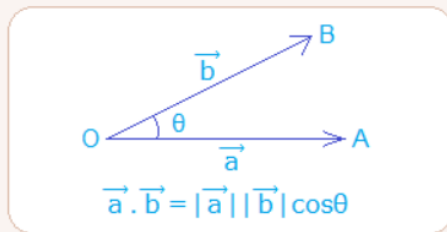
After completing most of the tasks in the unit, I found myself offering advice to other students on Discord, sharing insights gained from my experiences and understanding of the material. This process allowed me to reinforce my own learning by revisiting concepts and explaining back to them. By helping others, I had to confirm my understanding of the math involved is correct, which in turn deepened my comprehension of the subject matter.

 **@Cabbage** This is the variables when I was adding them for reference 

 **Asol** Today at 6:43 PM
Bro it said unsupported operand type for + between a Vector and a int.
So it probably you are adding a Vector with an Int.
Which mean your target_pos or target.vel is an int (I bet it is target_pos)
If so, did you changed target_pos into int though length() function or so?

 **Asol** Today at 6:59 PM
I think the problem could be your target_time variable
It needs to be int so your (target.vel * target_time) remain as vector2D
How you calculate the target time?
I have found out this for you
When multiply vector with vector2D, it gonna be an Int (edited)

Product of Vectors



So since you need to add the target_pos with (target.vel * target_time), both of these must be vector2D (edited)

Conclusion

In summary, I believe that I have clearly demonstrate that my portfolio with requirement task is sufficient to be awarded a HD grade.