## horizontal line



Pathfinding

06.04.2022

**─**

Ryan “Lunar” Bassil

800406188

# 

# Contents

[horizontal line 0](#_Toc63322665)

[Contents 1](#_Toc63322666)

[Task #1 1](#_Toc63322667)

[Question #1 1](#_Toc63322668)

# Task #1: Research

## Find 5 major AI jargon of your choice used within the games industry. List and describe the major AI terms.

**Machine Learning**

Essentially a field of Artificial Intelligence, in a broad sense, that defines and develops a machine’s ability to imitate human behaviour in complex task learning.

**Data Science**

An interdisciplinary field that uses scientific methods, processes, algorithms, and systems to extract knowledge and insights from the data in various forms, both structured(tables) and unstructured (images, videos).

**Pattern Recognition**

It is the automated recognition of patterns and regularities in the data.

**Computer Vision**

A field of AI that applies machine learning to teach computers how to interpret data from captured images and video.

**Deep Learning**

A way for an AI, or machine, to practice machine learning in a way that imitates how humans gain certain types of knowledge and behaviours.

**AI Agent**

The entity that performs, executes, and behaves with the AI code and information. Within this project, it shall be a simple game object with the attached script.

## Describe what an algorithm is, and how do you use algorithms when programming? Define what is a search algorithm and a sorting algorithm.

Algorithms are essentially step-by-step methods or procedures that are used for calculations, data processing, and automated reasoning.

A search algorithm is a series of steps of calculations to locate specific data within a collection of data, such as finding a specific word in a book.

A sort algorithm is a series of steps of calculations to analyse all the data in a collection and organise it based on specific requirements, for example, organising all the words in the book from shortest to longest in length.

## What is big O notation?

Big O notation as a concept can be applied to many philosophical categories of life, however when talking about AI specifically it relates to the upper boundaries or the largest scale of the code or data within the AI. In other words, it is used to describe the computation of executing time or space requirements grow based on the size of the input, and generally speaking, the “Big O Notation” is account for the worst case scenario and the largest limit of data and processing within time and space for the AI in question.

## Outline the difference between 3 different pathfinding algorithms.

**Breadth First**: An algorithm that simply searches all path possibilities equally.

**Dijkstra**: An algorithm that eliminates searching paths that it deems unnecessary to search or less likely options to finding the shortest path to the objective.

**A\***: A modification of the Dijkstra algorithm, the A\* algorithm searches for the shortest path to the objective as possible but also accounts for calculating an estimation of how far the origin to the objective distance is and uses the distance from the origin to determine the best path to use.

## Describe how object-oriented programming and inheritance could be used to create AI.

Since OOP is a language where the programmatical objects simulate or relate to those in which the behaviours and states of real-world objects relate to each other, and inheritance is used to describe how children objects (or more specific objects) possess characteristics, data, behaviours, and state of their parent objects, then using these concepts with AI can prove very beneficial. Machine learning and AI is all about how a machine learns to imitate human behaviour and how humans learn that ability, and in a similar way, humans learn by way of relating real world and theoretical objects to each other and considering the differences, then storing that information for later. In this way, an AI can be taught to identify a parent object (a general concept or over-arching item such as ‘food’) and relate it to a child (or specific) object such as ‘chicken’. Using OOP and inheritance, it would be able to recognise that chicken is a child object of food, and would thus understand its base behaviours and state, but it could then learn and store chicken’s specific characteristics separate from its parent. I.E: what makes chicken different to food as a concept, or what makes chicken different to carrots.

## Outline the development process for creating AI strategies for NPCs in a game.

* Research existing AI in games
* Think up a concept for an AI behaviour
* Draw up a flowchart of behaviour for the AI
* Draw up a UML diagram for its structure
* Write pseudocode
* Implement code
* Test

# Task #2: Design your pathfinding project

## Create a TDD(Technical Design Document) describing the design of your project. Your project needs to contain the following:

## • Pathfinding between different waypoints within a maze.

## • At least 3 agent types running in different directions and different waypoints.

## • A changing maze with doors that open and close automatically changing the path between points.

## • Different area modifiers that will affect different agents and their pathfinding. (the Unity NavMeshModifier/volumemodifier)

## • A gap between two different points on the map requiring a link (nav mesh link)

## • There needs to be two types of “collectables” that the AI must grab in order to complete the maze, using a state machine to change whether it is attempting to continue through the maze or going to collect something. Examples of collectables are:

## o A key to some sort of door

## o Some sort of coins/treasure

# Task #3: Develop your pathfinding project

## Create your project based on the design in your previous task:

## • Comment your code including function summaries.

## • Use NavMeshComponents from Github to develop the pathfinding.

## • Use multiple scripts for different components of your game.

## • Creative commons animated characters for AI agents, adjusting their animation based on walking/stationary states.