Artificial Intelligence for Games – Project Research Workbook

## Game/Application/Simulation Research

[Primer’s Natural Selection simulation](https://www.youtube.com/watch?v=0ZGbIKd0XrM) ([interactive version by MinuteLabs.io](https://labs.minutelabs.io/evolution-simulator))

A simulation of natural selection over many generations with three distinct mutations: speed, size, and sense.

## Pathfinding Algorithm Implementation

A\* pathfinding will be used.

Resources:

* YouTube
  + [The Coding Train](https://www.youtube.com/user/shiffman) – [Coding Challenge 51.1: A\* Pathfinding Algorithm](https://www.youtube.com/watch?v=aKYlikFAV4k)
  + [Sebastion Lague](https://www.youtube.com/channel/UCmtyQOKKmrMVaKuRXz02jbQ) – [A\* Pathfinding (E01: algorithm explanation)](https://www.youtube.com/watch?v=-L-WgKMFuhE)
* [Nicholas Swift – Easy A\* (star) Pathfinding (Medium.com)](https://medium.com/@nicholas.w.swift/easy-a-star-pathfinding-7e6689c7f7b2)
* [GeeksforGeeks – A\* Search Algorithm](https://www.geeksforgeeks.org/a-search-algorithm/)

## How the Pathfinding Algorithm Will Be Used

Agents in the application will seek out sources to aid them in surviving – such as a food or a source of water. The simulation will contain herbivores, carnivores and omnivores; plants will be randomly generated for the herbivores and omnivores, whereas the carnivores will rely on other agents or carcasses for a food source.

Due to the nature of the simulation, a tile-based grid will be used to simplify random-generation and tying that in with the pathfinding algorithm.

## AI Strategies Used

1. Behaviour Tree

A tree-shaped (acyclical) decision making technique that iterates over nodes each frame. These nodes can be for anything from replicating common operations (and, or, while) to game-specific logic (e.g. navigate path, find player).

Resources:

* [Chris Simpson – Behavior trees for AI: How they work (Gamasutra)](https://www.gamasutra.com/blogs/ChrisSimpson/20140717/221339/Behavior_trees_for_AI_How_they_work.php)
* [Holistic3d – Introduction to Behaviour Trees (YouTube)](https://www.youtube.com/watch?v=uq8hnnkAxsw)
* [Unreal Engine – Unreal Engine AI with Behavior Trees (YouTube)](https://www.youtube.com/watch?v=iY1jnFvHgbE)

1. Blackboard System

A shared source of knowledge for AI units to use; additionally, can be used by AI to either submit or fulfil requests/questions, depending on their role in the overall system.  
  
Resources:

* [GDC – Nuts and Bolts: Modular AI From the Ground Up (YouTube)](https://www.youtube.com/watch?v=IvK0ZlNoxjw)
* [Nalla Senthilnathan – The Blackboard Pattern for Autonomous Navigation](https://dzone.com/articles/the-blackboard-pattern-for-autonomous-navigation)

## What is a Technical Design Document?

* The manifestation of thoughts and ideas from a team, written and planned.
* Something that can determine scope and milestones; this can assist in estimating timeframes.
* A place for goals and functionality to be clarified, with details on *why* and *how* a feature will be implemented.
* The place that a team references when discussing or implementing specifics of a project.
* Potential and current issues that arise can be written in this document for further inspection.

## External Components (Libraries, Assets, Algorithms, etc.)

Graphics will be done through [*Raylib*](https://raylib.com/), this is a graphical abstraction layer built on top of OpenGL.  
Physics will be implemented with the [*Box2D*](https://box2d.org/) library.

The [*A\* algorithm*](https://en.wikipedia.org/wiki/A*_search_algorithm) will be used for pathfinding, alongside [*behaviour trees*](https://en.wikipedia.org/wiki/Behavior_tree_(artificial_intelligence,_robotics_and_control))*.*

Pre-existing sprites from [*Itch.io*](https://itch.io/) or [*Kenney.nl*](https://kenney.nl/) might be used, credit will be given in a text file with source.

## Licensing

[*Raylib uses the zlib/libpng license.*](https://www.raylib.com/license.html)

[*Box2D uses the MIT license.*](https://github.com/erincatto/box2d/blob/master/LICENSE)

[*Kenney assets are under the CC0 1.0 Universal Public Domain license.*](https://creativecommons.org/publicdomain/zero/1.0/)

## Possible Audience

Target platforms: Windows (x86\_64), Mac OS (x86\_64 *untested*, ARM), Linux (x86\_64 *untested*)

## Description of the Real-World Environment Simulated

The environment will be a very basic implementation of generic animal behaviour – both predator and prey.

Behaviour includes searching for food, searching for water, fleeing predators and following prey.

## Complexity

Behaviour trees seem moderately complex, even more so when trying to navigate many units with A\* on a tile-based grid.

Classes needed: (estimate: quite a few)

* 1 for behaviour tree, 1 for basic nodes, 1 per action (specific to game)
* 1 for pathfinding
* At minimum 1 for physics
* 1 or more for implementing basic GameObject functionality, with translation and rotation
* 1 for sprites, potentially a second inherited for animated sprites

## Tools For Assistance

Microsoft’s Visual Studio IDE will be the main application used for development and debugging for Windows; JetBrain’s Clion will be secondary to ensure proper compilation on Mac OS.