Documentation for AI practical project

S202447 (Stefan Simtea)

*Project codename: SHAP (Stefan’s horrible AI project)*

# Getting started

The goal of this project is to showcase on a very basic level some AI functions that we have gone through in the module. This is in no way advanced and lacks a lot of features but strives to showcase Finite State Machines, Path finding and path following and Steering Behaviors.

# Controls

By pressing “Space” you can get the details about the current state which should update every 2 seconds.

# Finite state machines

Text

Description automatically generatedIn the project there can be found 9 files related to Finite State Machine:

* BaseState.h – which is a template that all the states use
* IdleState.h and .cpp – which represents the idle stage, a default starting stage which can dictate when to enter State One or Two depending on a value given (in the project that value is time in seconds)
* StateOne and StateTwo .h and .cpp are some example states that I have made to showcase the functionality of the project. They always return to IdleState therefore IdleState dictates which state is selected.

# Steering Behavior

# The program uses different agents that move around seemingly at random to demonstrate steering behavior. They tend to distance themselves from others. They are being pushed around by forces and are being updated in the Update member function of the BaseEntity.cpp

# Text Description automatically generated

Most of the logic for the boid movement has been reproduced from the ones made in class during the module with a few adjustments.

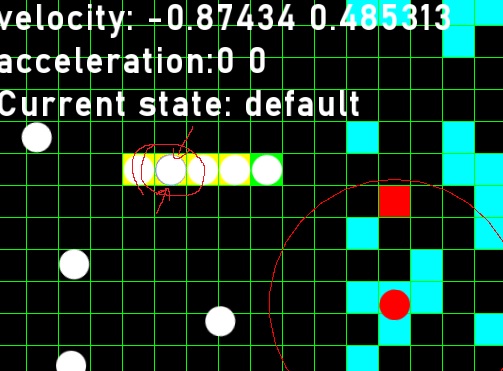
# Path finding

The program uses the A\* along with the Manhattan distance to calculate the fastest way possible from one cell to the other avoiding the cells that it should not travel through.Text

Description automatically generated

The code above demonstrates the check that the algorithm goes through in order to omit the cells that are not marked as “blank space” (type 0 is blank space, type 1 is a wall)

# Path following

Path following is a little more interesting, it is done in a very peculiar way. After I find a path that I want to assign to the Path following agent, I put boids onto that path on every cell and then I make the agent follow each one of them every 2 seconds until it reaches the end. Why this approach? It all boils down to lack of time, not because there was not enough time, just because I always find myself working near a deadline. I was doing progress throughout the whole module otherwise I could not have done it all these couple of days. I just left path following and behavior trees (which are not present) for last.

The boid that is following the path can be seen moving behind the boids that are on the path, you can sort of see a small little blue outline there where I circled with red.

# Object oriented programing

The project is using OOP for almost all the systems (not as much as I wanted in the end, but a fairly decent number)

The finite state machine implementation is done using objects and classes that way it gave me a neater approach to it’s implementation (as seen in code):





(main.cpp)

The boids are also classes that derive from BaseEntity and their logic is mostly handled in their class functions.

The path follower is another example, It is a class that derives from BaseEntity and handles it’s logic inside the class. The main function contains only the initialization of an object of PathFollower type and a function call.

# Bibliography

Even though the project may seem lackluster, research has been done. I just struggle with implementing most of it.

Sfml-dev.org. 2021. *sf::Clock Class Reference (SFML / Learn / 2.5.1 Documentation)*. [online] Available at: <https://www.sfml-dev.org/documentation/2.5.1/classsf_1_1Clock.php>

Sharma, S., 2021. [online] Youtube.com. Available at: <https://www.youtube.com/watch?v=1vfhlozfCRk&t=317s> .

Sharma, S., 2021. [online] Youtube.com. Available at:< [C++ & SFML | Open World RPG [ 175 ] | Artificial Intelligence!!! Let's start with a few components.. - YouTube](https://www.youtube.com/watch?v=fSra58fzvRo)>.