Second Assignment
Computer Game AI
COMP09041

Issue Date: Tuesday, March 14th, 2023
Due Date: **5pm, Tuesday, April 11th, 2023**

OpenAI Non-Player Characters in Raylib (Pt. 2)

In this assignment you should continue to develop a retro 2D role-playing game (RPG). You can either continue to work with the source code submitted for the first assignment, or you can make a fresh start with the updated source code you have been provided for this, second assignment.

The code provided with this second assignment includes an `NPC` class which means that multiple NPCs are straightforward, and code repetition can be completely avoided. The `NPC` will also communicate with OpenAI using `std::async` and `std::future`, ensuring there is no break in gameplay while awaiting a server response. Support for the "Tiled" application is also provided, allowing maps to be generated from sprite tilesets using the GUI provided by "Tiled". The

rudimentary *ashlands.tmx* file provided can be loaded in "Tiled" and used to export the *ashlands.json* file loaded by the raylib-tileson library. There is also now a *player* character class (`PC`) which avoids the need for multiple `Sprite` objects (i.e. up/down/left/right).

You are also free to integrate A* pathfinding, or computer vision (e.g. flow), from recent seminars into the gameplay.

Work in the groups you have been assigned to. The group leader should submit the assignment upon completion as before; though note that there is one part of the assignment which must be completed individually (see below). You are again encouraged to experiment; and have fun with the features of the OpenAI API.

Peer Review

As this is a group project, a system of peer review will be introduced to ensure that those who have worked hard are rewarded. When you submit your asiignment, **each of you** should send an email listing the names of individuals in your team (including yourself). Beside each person's name, you should put a mark out of ten $(0 - 10)$. For example:

Erika Mustermann (myself)	8
Mario Rossi	6
Joe Bloggs	1

Marking Scheme

The assignment is worth 40% of the marks awarded for the entire COMP09041 module. The following provides a breakdown of the marking scheme:

Game Design Document (1 or 2 pages)	5
NPC Interaction (which affects the game)	5
Interesting Gameplay	10
Overview Video with Dialogue (60 secs.)	5
Game Visuals	5
Code Quality (e.g. C++ classes; medium-sized functions)	5
Peer Review (averaged from your team mates)	5

Group Submission

Submit one zip file, per group, by the deadline noted above. Include your game design document; your C++ source files; and any resource files required (e.g. audio or image files). You can either include the overview video as a file within the zip, or provide a link to it in the game design document (GDD). You do not need to include your CMake build directory in the zip. Also remember to send the individual peer review email to *paul.keir@uws.ac.uk*.