**Assignment 1**

**Game Programming 2**

**COMP 396**

**(Individual Work)**

**Description**

The purpose of this assignment is to make an intelligent enemy for the 3D maze game in Unity.

In the 3D maze game, the player needs to find the exit while facing different enemies. We want to make an intelligent enemy who can patrol the area in which it is spawned. If the enemy sees the player its state is turns to chase state and if it gets close enough its state turns into attack state. Our enemy by default has some default ammunitions. If during the attack its ammunitions run out, the enemy state turns into the flee away from player. If player stops chasing the enemy, the enemy stops and looking around, i.e. the position is fixed, until it gets refilled after certain period of time. Note that if the enemy has enough ammunitions its state backs to attack if still sees the enemy and is close enough otherwise it goes back to the patrol state again. Graphically, our state machine looks like this:

Chase

Attack

Patrol

Start

Moving to target

Find the next target

Sub state machine

Start

Fire

Sub state machine

Flee away from player if is chased

Stops and looking around if is not chased

**Requirements**

1. You should implement the state machines using Animator in Unity
2. Define appropriate parameters for your state machines and use them for transitions.
3. Every state might have a behavior class which controls the exit, enter, and update of the state.
4. At least 4 different enemies should be spawned in the maze patrolling different regions.
5. To show the functionality of your state machine, exceptionally for this game, neither player or enemy will die. They only run out of ammunitions and they will be reloaded after different amount of time for the player and enemies.

**Marking scheme**

1. Finished working all 3 state machines for the defined maze game based on the requirements worth 80 marks.
2. Interesting maze with interesting animation effects for different states worth 20 marks.
3. **Note that your submissions shouldn’t have any compile or compatibility problem. In other words, after downloading, it can be opened directly in Unity 2018.2.xxx without any problem. If there is a problem of opening of the file, there is no guaranty that your assignment will be marked.**

**Submission**

Include all required assets in your project and compress it using zip and submit it before the due date.