COMP305 – Game Programming 1

Mid-Term Exam

Complete The 2D Tank Game

Due Week #7 (March 2, 2016) by end of class

Value 15%

Complete The 2D Tank Game **Maximum Mark: 24**

Overview: You will complete as much of the **2D Tank Game** as you can. The 2D Tank Game has several Red Enemy Tanks rolling toward the Player's Green Tank. The Player will use mouse or keyboard controls to avoid the Enemy Tanks. Several Scripts, Prefabs and Sprites have been included for your use along with a Base Unity Project.

Instructions:

(20 Marks: Functionality, 4 Marks: Version Control)

1. **Task: 1** Attach the Scripts to the appropriate GameObject. You will need to ensure that the public properties of each script showing up in the inspector are modified in order for the objects to operate correctly. (8 Marks: Functionality)
    a. Attach the **GameController** Script to the **GameController GameObject**. Ensure that the **GameController GameObject** instantiates at least 4 enemy tanks (1 Mark: Functionality).
    b. Attach the **EnemyController** Script to the **Enemy GameObject**. Ensure that the movement boundaries and speed of the enemy are input in the inspector appropriately. (2 Marks: Functionality).
    c. Attach the **PlayerController** Script to the **Player Object**. Ensure that the movement boundaries are adjusted in the inspector and that a reference to the **Camera GameObject** is attached so that mouse movement will function. Player movement will work with the mouse or keyboard input (your choice) (3 Marks: Functionality).
    d. Attach the **Road Controller Script** to the **Road GameObject**. Ensure that the speed is set to 3 in the inspector. (1 Mark Functionality).
    e. Apply all the changes to each of the Prefabs as required (1 Mark: Functionality).
2. **Task 2:** Create a Scoring mechanism for the Player. Each time the Player GameObject avoids a tank and the tank reaches the **bottom of the screen** the player will receive 10 points. The player starts with 5 lives. Display the score in several **Text** objects. (4 Marks: Functionality).

3. **Task 3:** Write a **PlayerCollider Script** and attach it to the **Player** GameObject. Ensure that when the Player collides with the tank that the tank object is Reset. The Player will lose 1 life. Add any missing components to the Player GameObject (4 Marks: Functionality).
4. **Task 4:** When the **Player Object** Reaches 0 Lives the Game is over. Write or modify a script to show the Player's high score and give the Player an option to restart the Game (4 Marks: Functionality).
5. **Task 5:** Share your files on **GitHub** to demonstrate Version Control Best Practices **(4 Marks: Version Control).**
    a. Your repository must include **your code** and be well structured (2 Marks: Version Control).
    b. Your repository must include **commits** that demonstrates the project being updated at different stages of development – each time a major change is implemented (2 Marks: Version Control).

BONUS OPTIONS:

1. This exam will be conducted like a **Game Jam** – The first person who completes the core requirements of the exam will gain a 5-point bonus
2. Add your own sounds for collisions, points, engine noise, etc. (4 Marks: Functionality).
3. Add your own soundtrack for the Game (2 Marks: Functionality).
4. Add a **Boss Tank** that appears after an allotted timeframe. The Boss Tank should be larger than the standard Enemy Tank and be displayed in another Colour (4 Marks: Functionality).
5. Ensure that enemy Tanks do not cross over / intersect with other enemy Tanks. (4 Marks: Functionality).
6. Empower enemy Tanks to **Fire bullets** at the Player. The Player will lose 1 Life every time every time the Player Tank is hit by a bullet (4 Marks: Functionality).
7. Empower the Player to **Fire bullets** and destroy the Enemy Tanks. Each time an enemy Tank is destroyed by a Player's Bullet the player's score increases by 50 points (4 Marks: Functionality).
8. Add a simple **Explosion Animation** that will be Instantiated when a Player is hit by an enemy, an enemy bullet or when the player's bullet hits an Enemy GameObject (4 Marks: Functionality).

SUBMITTING YOUR WORK

Your submission should include:

1. A zip archive of your Unity Project files uploaded to e-centennial.
2. Provide a link to your repository on GitHub.

Please zip all files in to a single project archive.

This assignment is weighted **15%** of your total mark for this course.