

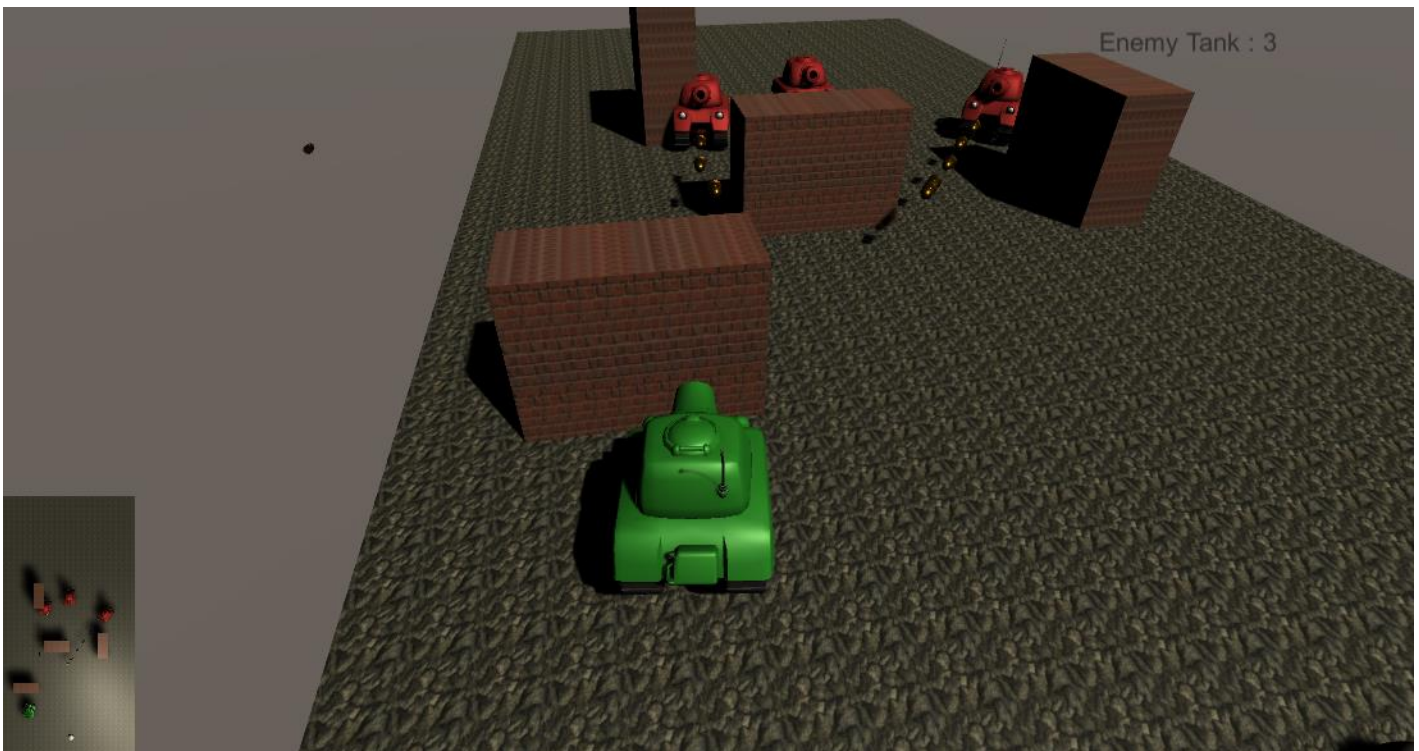
CSCI4120 Principle of Computer Game Software

Assignment 2

Tank Game (FSM Scripting & Homing Missiles)

Due: Apr10, 2020(Friday)

Scripting refers to the techniques of writing scripts for pawns in 3D games so that the pawns will have desired behavior according to the game design. In this assignment, you are required to implement a finite state machine (FSM) to control some tanks in Unity game engine with various behaviors, and with a special weapon being built for the player.



Requirements

You can use the level built according to our tutorial document to continue the work in this assignment.

You should :

1. write the scripts for two different behaviors to apply on two enemy tanks.
2. write the scripts to produce a formation based weapon for the player.

I. Finite State Machine

The tanks should have *common* behavior (in addition to the working style later) as follow:

- 1) The tank should scan for whether the player is nearby continuously. If the player is visible within a distance of 30 units, then the tank should enter attack mode. The scanning process should be able to :
 - I) detect the player within its view cone of angle 120 degree forward with 30 units distance,
 - II) check visibility of the player through an unobstructed line of sight i.e. no walls or other objects in between.

- 2) In attack mode, the tank will first check whether player is within attack distance (30 units). If yes, it will then orient itself towards you, and fire a missile at you. Then it will wait for 1 second and repeat step 2. Otherwise eg. Enemy tank leave, the tank will return to normal(original working state) mode.

In addition, the tanks should have two different styles of working.

- 1) Goal seeking

A goal point is set in the level (the flag of the player). The tank should proceed towards the goal point.

- 2) Patrol

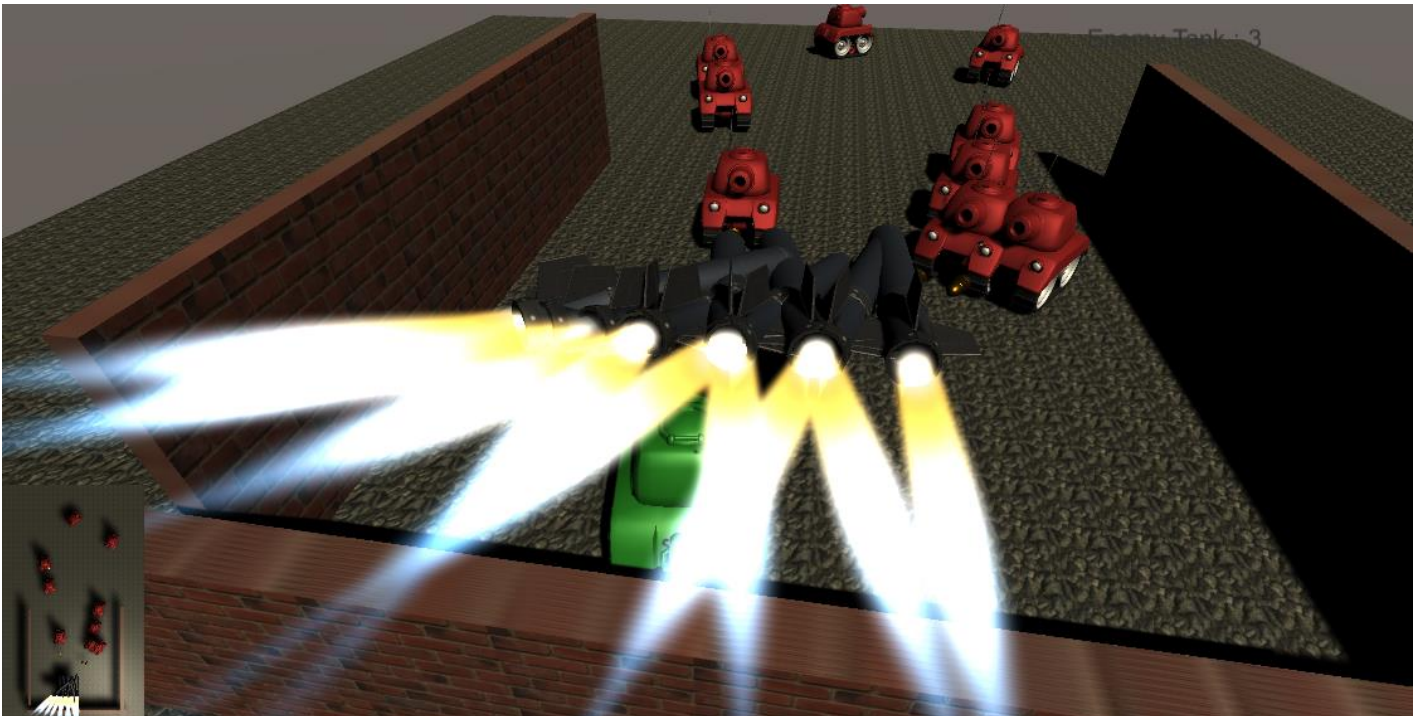
Three waypoints will be set up in level (ordinary GameObject with collider). The tank should patrol between these waypoints.

Each tank should have a life point of **300**, and with each hit by missile, no matter from friendly fire or player, a reduction of 10 life points will be deducted. A zero life point will see the *removal* of the tank from game.

The fired shell should also be removed from game if they collide with tanks or player tank. However you need not handle those fired shells which did not collide with tanks.

II. Homing Missiles

Homing missiles are very cool in their actions. We would like to implement a homing missile in this part. The missile should be able to target at a moving target i.e. the missile should update its course of direction according to the position of the target. In this case, the missile need to update its direction according to the target's position.



Your implementation should be as follow.

The homing missile will be triggered when player press *right mouse* button. Once it is pressed, the player tank should check all the current enemy tanks within a radius of 50 unit distance. Now for all those tanks within range, a homing missile should be created with its corresponding target, one missile per tank.

On colliding with the target tank, the target tank should be eliminated immediately no matter the life point is.

Remark

1. This assignment need you to use extensive scripting and physics engine knowledge. Unity official scripting documentation should help.
2. You can learn a lot on Unity by following this nice tutorial series
https://www.youtube.com/watch?v=WNmT6U74Lcc&index=6&list=UUGnWvk218j_hRgBICrbLtYg
3. Waypoint in Unity
<http://forum.unity3d.com/threads/a-waypoint-script-explained-in-super-detail.54678/>
4. Game object rotation is a relatively difficult issue. In this assignment, you are not required to handle them cleanly. We will only focus on the actions taken by the tanks.
5. Homing missiles are very interesting to implement though it looks a bit complicated at start. You may consult the following tutorial by “Dapper Dino” for details.
https://www.youtube.com/watch?v=t_-kTapjfhM

Submissions

Submit your completed scripts together with other needed files (zip the complete project folder) in a single archive to your cloud drive, and send us the public link so that our tutor will be able to download and test. Remember you are required to submit the link to our Blackboard assignment page on or before the deadline.

<http://blackboard.cuhk.edu.hk>