**RobotBattle README**

---------------------------------------------------------------------------------------

*Introduction*

Hello and thank you for joining the Robot Battle! This text will explain how you can set up an AI for your robot by using the provided framework. It will also explain certain rules, to make sure that the battles will be fair.

*Robot parts and functionality*

For this game, everyone will be using the same robot. This robot consist of multiple components:

**Wheels**: This is the bottom part of the robot. It is able to move forwards and backwards, and rotate left and right. The speeds for this has a maximum.

**Gun:** This is the middle part of the robot. It is a turret that can rotate separately from the wheel. It can shoot a bullet in the direction it is currently aiming at. The shooting speed and rotation speed have a maximum.

**Scanner:** This is the top part of the robot. It will scan the area it is looking at to see if there are other robots nearby, which will give you certain information about these robots. It can also rotate separately from the gun and wheels.

*How to set up your own robot*

Start by creating a C# script, and give this script your own firstname and lastname with an underscore in between (E.G. John\_Smith). This is to prevent that there are multiple scripts called ‘TankAI’ during the battles. The script will automatically be attached to a tank prefab when you run the game in one of the game modes (explained later).

To make this script functional, it needs to inherit RobotControl, and start with a protected override Start function. Therefore the start of the script will look like this:

public class John\_Smith : RobotControl {

protected override void Start()

{

{

       base.Start();

}

void Update ()

{

}

}

*API*

Because the script you will be using is inheriting RobotControl, you will be able to use certain functions to control your robot. These functions are also the only way you are allowed to control your robot, or to retrieve information from other robots. Look at the exampleAIs in the Unity Project to see the functionality. The functions will be explained here as well:

**MoveRobot(float movePower)**

This will move your robot forward or backward. movePower has a value between -1 and 1. Using 0 will not move your robot, negative numbers will move it backward and positive forward.

**void RotateRobot(float degree)**

This is used to rotate your robot. Degree is the number your robot will rotate towards. There is no limit for this, even though rotation would not exceed 360 degrees (This means to rotating from 0 degrees to 400 degrees will only rotate is 40 degrees). Degrees are counted clockwise.

**void RotateGun(float degree)**

This function will rotate the gun of the robot. The function is the same as RotateRobot(). The gun does not automatically turn when the wheels are turned.

**void RotateSensor(float degree)**

This function will rotate the sensor of the robot. The function is the same as RotateRobot(). The sensor does not automatically turn when the wheels are turned.

**void Shoot(int index)**

This will shoot a bullet in the direction that your gun is facing. It has a cooldown of 0.5 seconds. The index will indicate which bullet will be shot. So far the only available bullet will be 0.

**Void OnWallCollision()**

This is a function you can create yourself. It is activated when your tank collides with a wall. NOTE: make sure to override this function in your own script.

**Void OnTankCollision()**

This is a function you can create yourself. It is activated when your tank collides with another tank. NOTE: make sure to override this function in your own script.

**Void OnBulletCollision()**

This is a function you can create yourself. It is activated when your tank is hit by a bullet. NOTE: make sure to override this function in your own script.

**Void OnVictory()**

This is a function you can create yourself. It is activated when you win a match. It allows you to do a victory-dance of some sorts. Purely aesthetic and for fun.

**AccessData MyData()**

MyData is of type AccessData. It contains the variables of your own tank that you are allowed to use for coding your own tank. The variables are listed below.

**AccessData[ ] FindTanks()**

FindTanks is an array of type AccessData. This array will store multiple variables that belong to any tanks you are currently scanning, which are listed below. These are the only variables you may use for reading data from another tank.

**Void ChangeColor(Color robotColor)**

ChangeColor allow you to change the color of your tank. This is purely aesthetic, to make your tank a bit more personal.

**Void ChangeColor(Color gunColor, Color bodyColor, Color wheelColor)**

ChangeColor allows you to change the color of your tankparts. This is purely aesthetic, to make your tank a bit more personal.

*AccessData variables*

AccessData contains multiple variables about your tank or other tanks. These are the variables:

**Float Health**

The current health of the tank.

**Vector3 Position**

Contains the worldposition of the tank.

**float Distance**

The distance between you and the other tank.

**float MoveSpeed**

The movement speed of the tank in units per second, set between -3 and 3. (NOTE: You control your own speed with -1 to 1. This is amplified by 3 to give the actual speed.)

**float RobotRotation**

The rotation of the wheels in degrees.

**float GunRotation**

The rotation of the gun in degrees.

**float SensorRotation**

The rotation of the sensor in degrees.

*Coding rules*

To make sure that everything is balanced, there are rules to what you can and cannot do with your code.

**Controlling your robot**

For controlling your robot, you are only allowed to use the functions mentioned in the API. This means you cannot manually move your robot with own code, like transform.Translate(). You may also only use Shoot for shooting projectiles.

**Retrieving data from your robot or other robots**

You may only retrieve data from another tank by using the AccessData, which is mentioned earlier. This means you cannot read any other information by getting information from their components.

You may also only use the scanner to find other tanks. This means you will not be able to find the other tanks from the hierarchy.

So code like this is not allowed: GetComponent, GameObject.Find, Transform.getChild, RayCasts, CheckBox, OnCollisionEnter, Instantiate or anything similar.

**Scripts**

You may only work in your own script. You may look in other scripts for reference (exampleAIs are even recommended), but you may not make any changes to them. Make sure all functionalities you create are contained in one script.

*Game modes*

If you start in the MenuScene, you will have three options: Versus Mode, Manual Mode, and Competition Mode.

**Versus Mode**

This mode allows you to practice a 1v1, which is the same as the battle during the competition. You can select 2 AIs and have them duel each other.

**Manual Mode**

Manual mode allows you to add up to 8 AIs to a battle. They will spawn in a circle, and the last tank standing wins.

**Competition Mode**

This is the mode we will use for the final competition. It will automatically set up a competition, and the victors will continue to the next round.

*Submitting your AI*

All that is needed from each person is their script for the tank AI. You are able to upload it in the folder on drive. They will then be added to one Unity Project, which will be used for the final battle.

*Battle setup*

All battles will be 1v1. Your character will spawn in a corner of the map, facing a wall. Your opponent will spawn in the opposite corner. We will always use the same map. (NOTE: Your tank might spawn with a rotation of 180 degrees. If you do not do anything to your rotation, the tank will automatically rotate to 0 degrees, since this is default in the script.)

The tournament is style is knock-out. This means that the winner will move on to the next round, and the other player will be eliminated from the tournament. All rounds are best of 3, and the finale is best of 5.

After this tournament there will ofcourse be an opportunity to challenge other people.