



Unity Piscine - D07

Navmesh, 3D Physics, 3D GUI

Staff staff@staff.42.fr

Summary: This is the D07 subject for the Unity Piscine of [42](#).

Contents

I	Preamble	2
II	Instructions	3
III	Today's specific rules	4
IV	Exercise 00: Draw me a sandbox	5
V	Exercise 01: One tank to rule them all	6
VI	Exercise 02: Wargames	7
VII	Exercise 03: Pimp My Tank!	8

Chapter I

Preamble

Time to go on a mission. Riding your tank, you will enter enemy territory and find a way to take them by surprise. Yes, today, we're gonna pull inspiration from **Worlds of Tanks** or other similar games. As usual, you're invited to check those fine demos to find out what today will be made of.

And here is a little track that will carry you through this perilous journey. A real earworm you won't be able to get rid of. Don't thank me.

Good luck!

Chapter II

Instructions

- Only this page will serve as reference. Do not trust rumors.
- The exercises have been ordered from easiest to most difficult. Under any circumstance you can submit or take into account an exercise if a previous one has failed.
- Be careful with the access rights of your files.
- You should follow the submit procedure for all you exercises.
- Your exercises will be corrected by your piscine peers.
- You cannot leave any extra file on your repository except the ones explicitly specify on you subject.
- Got a question? Ask your peer on the right. Otherwise, try your peer on the left.
- Everything you need can be found on the `man` or out there on Google.
- Read carefully the exercises: they may contain some features you should implement that are explicitly mentioned on the subject.
- Think about discussing on the forum Piscine of your Intra!
- Use your brain!!!


Chapter III

Today's specific rules

- [Freedom](#)

Chapter IV

Exercise 00: Draw me a sandbox

	Exercise 00
Exercise 00: Draw me a sandbox	
Turn-in directory : <i>ex00/</i>	
Files to turn in : *.cs	
Allowed functions : Anything but what's specified in the instructions	

Now you got your degree in terraforming, you're gonna be able to design a wonderful terrain. Use your imagination. The theme must match the game type... remember tanks and unicorns are quite different.

Here are a few instructions you'll have to observe for your playground:


- The size will be 256x256.
- Try to split it into different areas with places you can hide and various heights.
- Use various textures. Enjoy yourself.
- You're free to choose your environment. The provided example is... just an example.



Think about your terrain before you build your NavMesh. Avoid terrains that are too steep or gritty.

Chapter V

Exercise 01: One tank to rule them all

	Exercise 01
Exercise 01: One tank to rule them all	
Turn-in directory : <code>ex01/</code>	
Files to turn in : <code>*.cs</code>	
Allowed functions : Anything but what's specified in the instructions	

Now you've got your sandbox, we're gonna be able to have fun adding toys into it. Create a tank. You must be able to move the tank's body forward with W and backward with S, and make it turn left with A and right with D. You can also rotate the canon with the mouse. The canon can only rotate on the Y axis and aim in the direction of the mouse's cursor (follow the example, it's pretty explicit).

You must also be able to use a boost and make the tank move faster pressing and holding the left Cap key. The boost is limited and must reload after a given time using it. What would a tank be without submachine guns and missiles? Add the ability to fire a submachine gun with the left click of the mouse. Right click will be used for the missiles. Missiles have limited ammo. After they're shot, you can't replenish them. Submachine guns and missiles must create particles where they hit.




For the submachine gun and missiles, you must use a Physical Raycast that will start in your canon and have a defined range.

Make sure you set sounds in your game (machine gun, missiles, explosions, tank engine, music...) to make it more lively.

Chapter VI

Exercise 02: Wargames

	Exercise 02
Exercise 02: Wargames	
Turn-in directory : <i>ex02/</i>	
Files to turn in : *.cs	
Allowed functions : Anything but what's specified in the instructions	


Now we have our toys, it's time to add challenge to the equation. And what better way to do so than add enemies? Create an artificial intelligence that can detect the closest enemy in range and attack it. "In range" means at the right distance and height to get hit. You've guessed right: you're gonna need to add life points to tanks. When the tank loses all its points, it explodes and disappears. If your tank explodes, the game ends and starts over.

A few technical considerations:

- Of course, you will use a NavMesh to move the tanks controlled by the AI.
- There must be at least 2 enemy tanks on the map and they can attack one another.
- The AI must not shoot regularly and can miss its target.
- You don't have to deal with the terrain obstructing the AI's aiming. An AI can shoot a mountain just because you're behind. You won't be penalized.
- If it's well designed, the AI won't have its tanks bump one into the other.

Chapter VII

Exercise 03: Pimp My Tank!

	Exercise 03
Exercise 03: Pimp My Tank!	
Turn-in directory : <i>ex03/</i>	
Files to turn in : *.cs	
Allowed functions : Anything but what's specified in the instructions	

Let's make this whole thing real classy adding a GUI. But not just any GUI! A very nice 3D GUI! We're gonna display a few essential informations about the tank, like, the tank's life and the number of missiles remaining. If you want to add more infos, just go ahead. Set a little distinctive sign that will show the GUI is indeed in 3D, and put the 3D to use.

Also add a little crosshair sprite and make it change color if the player hits a target.