

COMP30019 – Graphics and Interaction

Lab 2

The University of Melbourne

Introduction:

In this lab you will be learning how to apply transformations to objects, and how to utilize basic user input to control game objects.

There are two files present in the project to start off with:

- *MainScene.unity* – A Unity scene containing a cube entity. This is the scene you’ll be modifying in today’s lab.
- *XAxisSpin.cs* – A C# script component which spins the object it is attached to about the x-axis.

Tasks:

- 1) Open *MainScene.unity* in Unity. Press the ‘Play’ button and take a look at the cube object – currently it will be spinning about its x-axis. Open the script *XAxisSpin.cs* to examine how this is done. Create 5 additional cubes which do the following:
 - a. Move linearly back and forward in the z-axis
 - b. Move linearly back and forward in the z-axis whilst rotating about the x-axis
 - c. Orbit about the x-axis
 - d. Orbit about the x-axis while locally rotating about the x-axis
 - e. Perform a ‘figure 8’ motion
- 2) Adjust the camera transform so that it views the cubes from above. All the cubes should be visible from the ‘Game’ tab when the simulation is running.
- 3) Write a script that allows you to move the original cube with the arrow keys on the keyboard (in the X-Z plane). You will need to use the `Input` class provided by Unity within the script. It might help you to look up the API for the `Input` class in the Unity documentation.
- 4) Write a similar script that allows you to *accelerate* the original cube with the arrow keys on the keyboard (in the X-Z plane). For example, pressing the right arrow will increase the velocity of the cube in the positive x-axis. Remember to disable the previous movement script component from the cube before attaching this one.
- 5) Write a script which switches the cube between using the movement script and the acceleration script when the ‘s’ key is pressed. Additionally, change the material of the cube such that it is rendered blue when the movement script is enabled, and red when the acceleration script is enabled.
- 6) Currently the cube can move outside the bounds of the camera view. Write a script which is to be attached to the camera, such that it tracks the cube from its position (i.e. by rotating). The script should theoretically allow any game object to be tracked, not specifically the cube. Note that it may help visually to add a static object to the scene as a reference point, such as a floor.
- 7) Change the camera to use orthographic projection. What is the difference between perspective projection and orthographic projection? Is there a notable *visual* difference?