UECS3273 Programming with Game Engines
May 2018
Quiz

This quiz constitutes **10%** of your overall marks
Answer **ALL** questions. The total mark is 50.

1. List the 5 main views that are found in the Unity main IDE window and briefly describe their purpose. (10 marks)

2. Describe in detail what a skybox is and how it is used. (6 marks)

3. State 4 guidelines that are useful to apply a terrain texture effectively. (8 marks)

4. Explain in detail how transforms work on nested objects with respect to their parents and their respective coordinate systems. (6 marks)

5. Explain the problem that unwrapped textures can help to solve, and how they solve this problem. (5 marks)

6. Figure 1 shows 6 Cube objects arranged into two rows. All the cubes in the top row have a positive value for their y coordinates, while the cubes in the bottom row have negative values. Each of these Cube objects are randomly assigned either the tag `Human` or `Animal`. When the game is played, the following functionality is observed:

- All objects labelled with the tag `Animal` will start rotating about the Y axis, but remain stationary
- All objects labelled with the tag `Human` will start moving on the Y axis. Objects from the top row will start moving downwards, while those from the bottom row will start moving upwards.
- When any two objects collide, both of them will disappear

There are two scripts in this game. `MoveObjectScript` (Figure 3) is attached to all 6 Cube Objects. Its function is to perform the animation of the objects in line with the requirements stated above. `FindObjectsScript` (Figure 2) is attached to the Directional Light and its function is to locate the appropriate objects with the specific tags and initiate the required behavior by calling the `startMovement` method in `MoveObjectScript`.

a) Complete the code for `FindObjectsScript`. Hint: The actual movement of the objects should be performed by `MoveObjectScript`. The code here only needs to determine which objects are tagged as `Animal` or `Human`, and then pass this information to a suitable method in `MoveObjectScript` so that it knows the appropriate animation to perform on the Cube object it is attached to. (7 marks)

b) Complete the code for `MoveObjectScript`. (8 marks)

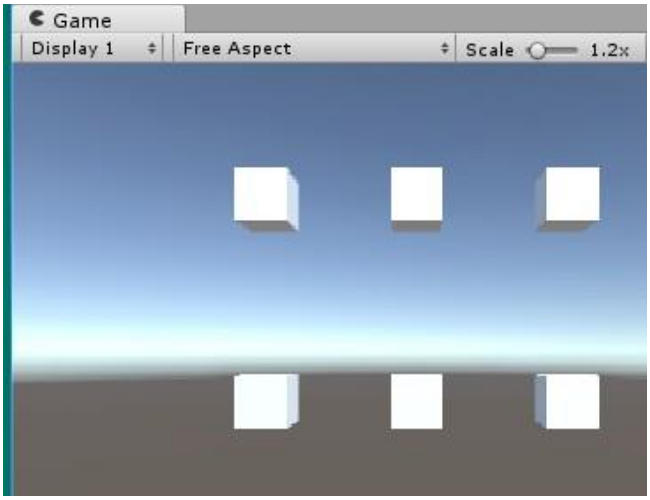

Figure 1

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class FindObjectsScript : MonoBehaviour {

      public GameObject[] animals;
      public GameObject[] humans;
      MoveObjectScript mos;

      void Start () {
                  // Complete code for a)
       }
      }
```

Figure 2

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class MoveObjectScript : MonoBehaviour {

      // used to determine whether the object rotates on the Y axis
      public float rotateY = 0f;

      // used to determine whether the object moves on the Y axis
      // and the direction of movement
      public float movY = 0f;

      void Update () {
                  // Complete code for b)
      }

      // the value passed to mytype from FindObjectsScript
      // will help determine the kind of animation to
      // perform
      public void startMovement(string mytype) {
                  // Complete code for b)
      }

      public void OnTriggerEnter(Collider other) {
                  // Complete code for b)
      }
}
```

Figure 3

ANSWERS

Q1. 10 marks

a) Project: Contains all the assets for a project (files, scripts, textures, models, and so on) organized appropriately into folders **[2]**
b) Hierarchy view: This shows all the items in the current scene instead of the entire project. **[2]**
c) Inspector view: enables the viewing of all the properties of a currently selected item in the Project or Hierarchy view **[2]**
d) The Scene View: provides a view of the various game objects involved in the construction of the game **[2]**
e) Game View: plays the game within the editor by simulating the action of the various objects shown through the view of the current active camera in the Scene view. **[2]**

Q2. 6 marks

A skybox is a cube consisting of six flat sides enclosing the world of the scene view **[2]**. It has inward-facing textures to make it look round and infinite. **[2]** Can use different skyboxes (or no skyboxes) on different cameras to make the world look different to different viewers **[2]**.

Q3. 8 marks

Any of 4 of the below
a) Try to make the pattern repeatable.
b) Try to make the texture larger
c) Make the texture square.
d) Try to make the texture dimension a power of two (4,8,16)
e) Keep effects subtle.
f) Elements fade from one element to another without harsh transitions

Q4. 6 marks

a) A child object's coordinate system is relative to the local coordinate system of the parent **[2]**
b) A nested child object's transform is relative to that of the parent object **[2]**
c) Transforms applied to parent object cause same effect in child, without actually changing the transforms of the child. **[2]**

Q5. 5 marks

Sometimes textures get applied incorrectly around a 3D model **[2]**. Unwrap is a map that shows exactly how a flat texture will wrap back around a model **[3]**

Q6. a) 7 marks

```
      void Start () {

            animals = GameObject.FindGameObjectsWithTag ("Animal"); [1]
            humans = GameObject.FindGameObjectsWithTag ("Human"); [1]
            foreach (GameObject animal in animals) { [1]
                  mos = animal.GetComponent<MoveObjectScript> ();[1]
                  mos.startMovement ("Animal"); [1]
            }

            foreach (GameObject human in humans) { [1]
                  mos = human.GetComponent<MoveObjectScript> ();
                  mos.startMovement ("Human"); [1]
            }

      }
}
```

a) 8 marks

```
      void Update () {
            transform.Translate (0f, movY, 0f); [1]
            transform.Rotate (0f, rotateY, 0f); [1]
      }

      public void startMovement(string mytype) {
            if (mytype.Equals ("Human")) [1]
                  if (transform.position.y < 0) [1]
                        movY = 0.01f; [1]
                  else
                        movY = -0.01f; [1]
            else
                  rotateY = 1f; [1]
      }

      public void OnTriggerEnter(Collider other) {

                  Destroy (this.gameObject); [1]

            }
      }
}
```