Programme Code: TU857, TU856, TU856, TU984

Module Code: CMPU 4030

CRN: 22528, 22418, 31083, 27945

**TECHNOLOGICAL UNIVERSITY DUBLIN**

**KEVIN STREET CAMPUS**

**\_\_\_\_\_\_\_\_\_\_\_\_\_**

BSc. (Honours) Degree in Computer Science (Infrastructure)

BSc. (Honours) Degree in Computer Science

BSc. (Honours) Degree in Computer Science (International)

**Year 4**

BA. (Honours) in Game Design

**Year 3**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

SEMESTER 1 EXAMINATIONS 2020/21

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Games Engines 1**

Dr. Bryan Duggan

Dr. Deirdre Lillis

Ms. Pauline Martin

Ms. Sanita Tifentale

Instructions to candidates

Answer Question 1 (Compulsory) and any 2 from the remaining questions

Question 1 is worth 40 marks, the remaining questions are worth 30 marks each

**Question 1**

A 3D tower defence game made in Unity has the following rules:

1. There is a crosshair in the centre of the screen that the player uses to select a location on the map in order to place a tower.
2. Towers become active when a creep comes in range.
3. When a tower becomes active it will turn to face the creep and continue targeting the creep so long as it stays in range.
4. Towers can fire 5 bullets per second.
5. Bullets disappear after 5 seconds if they don’t hit anything
6. There are three possible types of creeps and each has an equal probability of being spawned.
7. Creeps follow a path to get to the players base.
8. When a creep is hit with a bullet, it explodes and after a few seconds, sinks into the ground and gets removed from the scene.

Taking each of the rules above, how you would program them in Unity?

(8 x 5 marks)

**Question 2**

1. Discuss the relationship between the quantities of *force*, *velocity*, *acceleration*, *position*, *distance*, *time* and *mass* in relation to 3D computer games. In your answer include:
2. Units of measurement and representations for these quantities. (5 marks)
3. Equations that describe the relationships. (5 marks)
4. A description of how to update the state of a Newtonian physics particle with respect to time in a 3D computer game. (5 marks)
5. A *gravity gun* in 3D games allows the player to grab an object and hold it at a point in front of the camera. Explain in detail how you would implement a gravity gun effect in a Unity project.

(15 marks)

**Question 3**

1. Figure 1 shows an extract from a generative physics system that creates the caterpillar animat given in Figure 3.

void Awake()

{

float depth = size \* 0.05f;

Vector3 start = - Vector3.forward \* bodySegments \* depth \* 2;

GameObject previous = null;

for (int i = 0; i < bodySegments; i++)

{

float mass = 1.0f;

GameObject segment = GameObject.CreatePrimitive(PrimitiveType.Cube);

Rigidbody rb = segment.AddComponent<Rigidbody>();

rb.useGravity = gravity;

rb.mass = mass;

segment.name = "segment " + i;

Vector3 pos = start + (Vector3.forward \* depth \* 4 \* i);

segment.transform.position = transform.TransformPoint(pos);

segment.transform.rotation = transform.rotation;

segment.transform.parent = this.transform;

segment.transform.localScale = new Vector3(size, size, depth);

segment.GetComponent<Renderer>().shadowCastingMode = UnityEngine.Rendering.ShadowCastingMode.Off;

segment.GetComponent<Renderer>().receiveShadows = false;

segment.GetComponent<Renderer>().material.color = Color.HSVToRGB(i / (float)bodySegments, 1, 1);

if (previous != null)

{

j.autoConfigureConnectedAnchor = false;

j.anchor = new Vector3(0, 0, -2f);

j.connectedAnchor = new Vector3(0, 0, 2f);

j.axis = Vector3.right;

j.useSpring = true;

JointSpring js = j.spring;

js.spring = spring;

js.damper = damper;

j.spring = js; }

previous = segment;

}

}

Figure 2

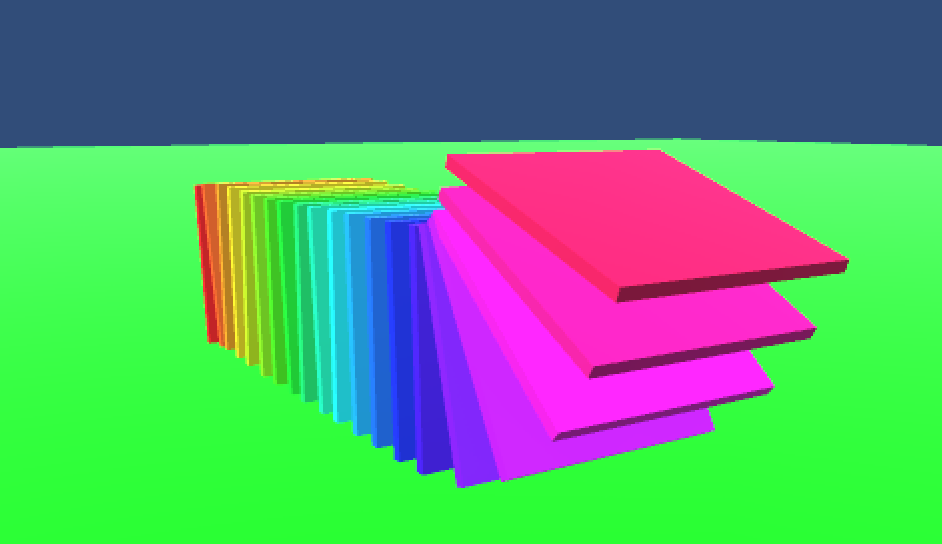


Figure 3

Answer these questions about the code:

1. How are the position, rotation, anchor points and scale of each segment calculated? Include a diagram in your answer

(10 marks)

1. How are the segments constrained to move relative to one another?

(10 marks)

1. How is the colour of each segment determined?

(3 marks)

1. What would the hierarchy look like after the Awake method has been called?

(2 marks)

1. How would you procedurally animate the caterpillar so that torque is applied to to each segment in sequence?

(5 marks)

**Question 4**

1. In relation to digital audio, explain the following terms: sample rate, resolution, frame size, spectrum, bin width.

(5 x 2 marks)

1. Figure 1 shows an extract from a Unity C# script that visualises the frequency spectrum of an AudioSource.

void CreateVisualisers()

{

float theta = (Mathf.PI \* 2.0f) / (float)AudioAnalyzer.frameSize;

for (int i = 0; i < AudioAnalyzer.frameSize; i++)

{

Vector3 p = new Vector3(

Mathf.Sin(theta \* i) \* radius

, 0

, Mathf.Cos(theta \* i) \* radius

);

p = transform.TransformPoint(p);

Quaternion q = Quaternion.AngleAxis(theta \* i \* Mathf.Rad2Deg, Vector3.up);

q = transform.rotation \* q;

GameObject cube = GameObject.CreatePrimitive(PrimitiveType.Cube);

cube.transform.SetPositionAndRotation(p, q);

cube.transform.parent = this.transform;

cube.GetComponent<Renderer>().material.color = Color.HSVToRGB(

i / (float)AudioAnalyzer.frameSize

, 1

, 1

);

elements.Add(cube);

}

}

// Update is called once per frame

void Update () {

for (int i = 0; i < elements.Count; i++) {

elements[i].transform.localScale = new Vector3(1, 1 + AudioAnalyzer.spectrum[i] \* scale, 1);

}

}

Figure 1

1. In relation to the code answer the following questions:
2. What shape will the generative visual have? How is the position of each segment in the visual calculated?
3. How is the orientation of each segment calculated?
4. How is the colour of each segment determined? What will the colour look like?
5. What aspect of the visual will be affected by audio? How would you improve the visual so that it was more responsive to the audio characteristics of music?