**TEMASEK POLYTECHNIC**

**SCHOOL OF INFORMATICS & IT**

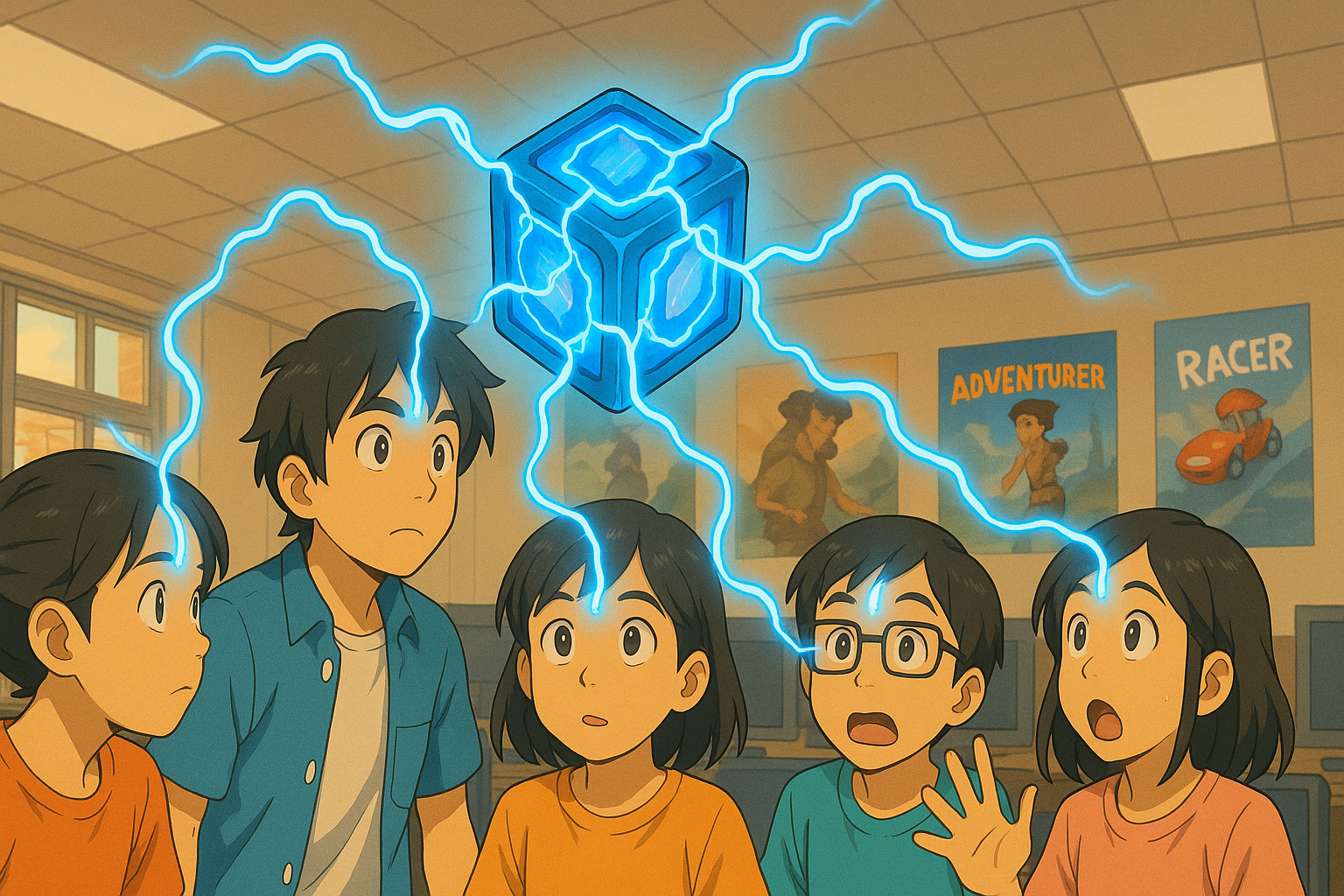
**DIPLOMA IN IMMERSIVE MEDIA & GAME DEVELOPMENT**

**AY2025/2026 APRIL SEMESTER**

**GADV (CGE2C25)**

**Unity Physics Worksheet**

To see the additional comments and resources, make sure you select **All Markup** in the **Review/Tracking** pane



**QUESTION 1 (basics of Unity physics)**

1. What is the core responsibility of a RigidBody component?

A RigidBody component simulates a solid, firm body in Unity.

1. Why is the Pong game paddle mentioned as an example of an object whose rigidbody’s isKinematic property would be set to true? Hint: what controls the paddle movement?

If the rigidbody’s isKinematic property is not set to true, the physics of the ball would affect the position of the paddle. The isKinematic property ensures that only the player’s mouse inputs can change its position.

1. Explain what the rigidbody **constraints** are for. Give ONE example of when this might be useful.

RigidBody constraints can be used to restrict and specify which axes are affected by the RigidBody physics. For example, using the Z rotation constraint prevents the player sprite from spinning off course when it bumps into a platform.

1. Watch [this video](https://www.youtube.com/watch?v=ixM2W2tPn6c).
2. What is one major problem with using the Translate function to move an object?

The object will ignore any form of collision or physics that are applied to it.

1. Why is the physics-based code put in the FixedUpdate function and not in Update?

FixedUpdate ensures that the code runs whenever any physics-based interaction should occur, while Update only runs every frame. Multiple physics interactions may occur in a frame, so it is purpose-built for such situations.

1. What are the THREE different methods covered in the video to move an object using physics?

Explain how each method works.

One of the ways is by using AddForce on the RigidBody. When used, the object speeds up before slowing back down to a stop, simulating friction as an external force.

Another way is by assigning a Velocity value to the RigidBody. In this case, the object maintains a constant speed, and is unaffected by any simulated external forces, like friction and acceleration.

There is also MovePosition. It behaves like Translate, but unlike Translate, it does not ignore collision or physics.

**QUESTION 2 (Colliders)**

1. What is the core responsibility of a collider component?

A collider detects when an object has collided with another object.

1. Look at the question in [this forum post](https://forums.oculusvr.com/developer/discussion/59641/problems-with-fast-moving-object-collision-in-unity-table-tennis-racket-vs-ball). What is the solution to the problem? Do some research to explain why this is the solution. Draw a diagram to illustrate your answer.

A common solution would be to change the collision mode on the racket and ball to Continuous Dynamic.

When the racket has its collision mode set to Discrete, it will only check for collision per frame. As such, if a collision occurs mid-frame, it will be ignored, causing the ball to go through it.

While Continuous might seem plausible at first, note that the ball is not a static object. Continuous may detect collision during mid-frames, but it only works on static objects with collision. The ball is most likely a RigidBody, thus it will be ignored.

Continuous Dynamic collision works on RigidBodies, so long as the RigidBody is not using the discrete collision mode as well. Therefore, the racket will collide with the ball if the ball’s collision mode is set to either Continuous or Continuous Dynamic.

1. What is the difference between collision *detection* and collision *resolution*?

Collision detection detects that two or more objects have collided with each other in the first place. On the other hand, collision resolution is code that ensures any collision error is handled correctly (e.g. Separating the objects or making them bounce off of each other should they clip into each other).

**QUESTION 3 (Joints)**

1. What type of joint is used in the video? According to Unity’s documentation, what does this joint do? Give an example of how the joint could be used in a game.

The **hinge joint** is used in the video. According to Unity, a hinge joint connects two or more RigidBodies at a particular point, [whilst still allowing the RigidBodies to rotate around the axis of that point.

1. What other types of joints are there? Briefly explain what FOUR of these joints does (do NOT include the type of joint you answered for part a above), and give an example of how each one might be used in a game.

Other Joint types include the **Configurable Joint**, **Character Joint**, **Fixed Joint** and **Spring Joint**.

Configurable joints are similar to those in a ragdoll, however they can be adjusted however way you like. Adjusting the movement of a configurable joint in a RigidBody can be used to make a skeleton for a character model. From there, you can limit them to your liking to make various animations and poses.

Character joints are similar to hinge joints, however they are only limited to angular and linear movements, as opposed to the free rotation of a hinge joint. These can be used on player character RigidBodies as well to give them arms and legs, though they are not as freeform as configurable joints.

Fixed joints are the least flexible of the joints. They are used to link two RigidBodies to each other, however they do not have any freedom to move or rotate independently. The two move along with each other. If you need swinging obstacles to swing along a consistent path (e.g. Swinging axe), fixed joints are perfect for that.

Spring joints links two RigidBodies in a similar fashion to a fixed joint, however the spring joints still allow them to move freely, as if they were attached to a string. The spring joints pull them together, much like tension in a real string. As such, these joints can be used for simulating rope or for animating weapons that contain string-like parts (e.g. flail).

**QUESTION 4 (Physic Material)**

1. What is the difference between dynamic friction and static friction?

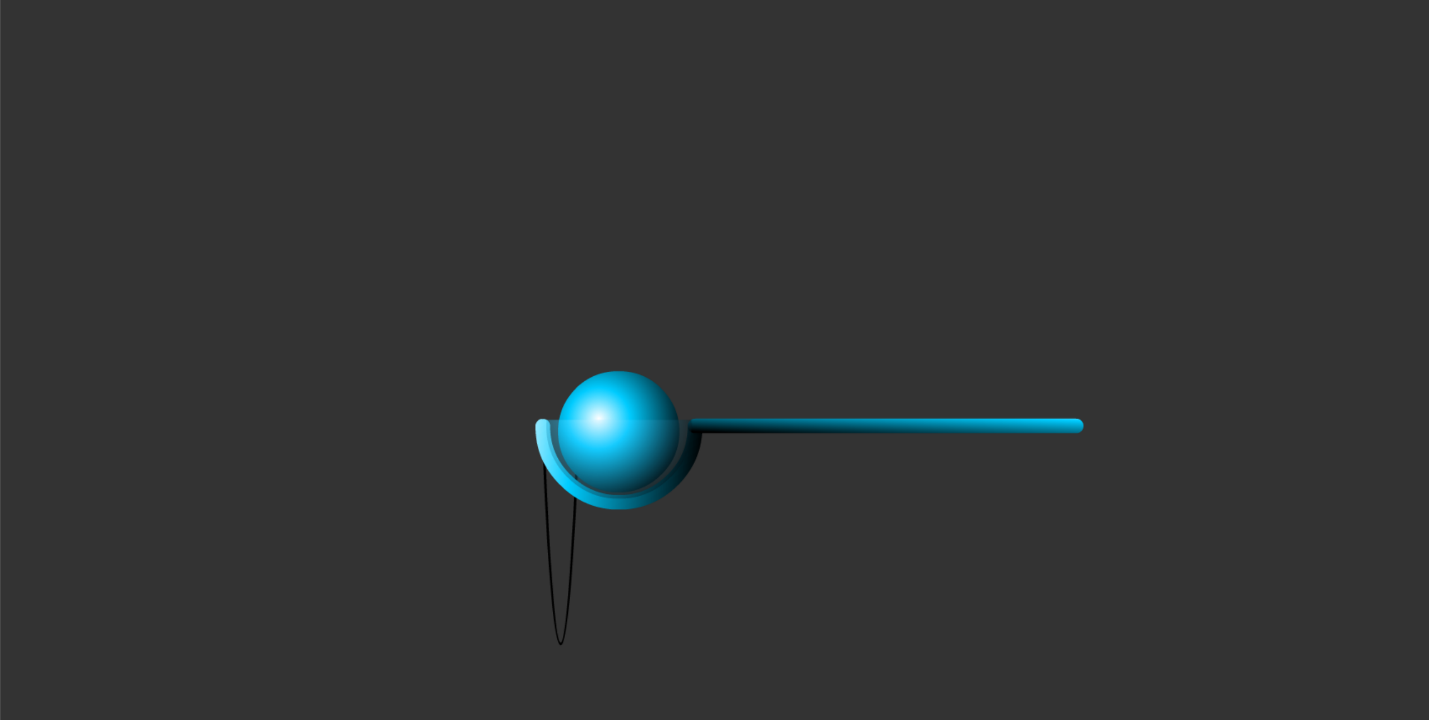
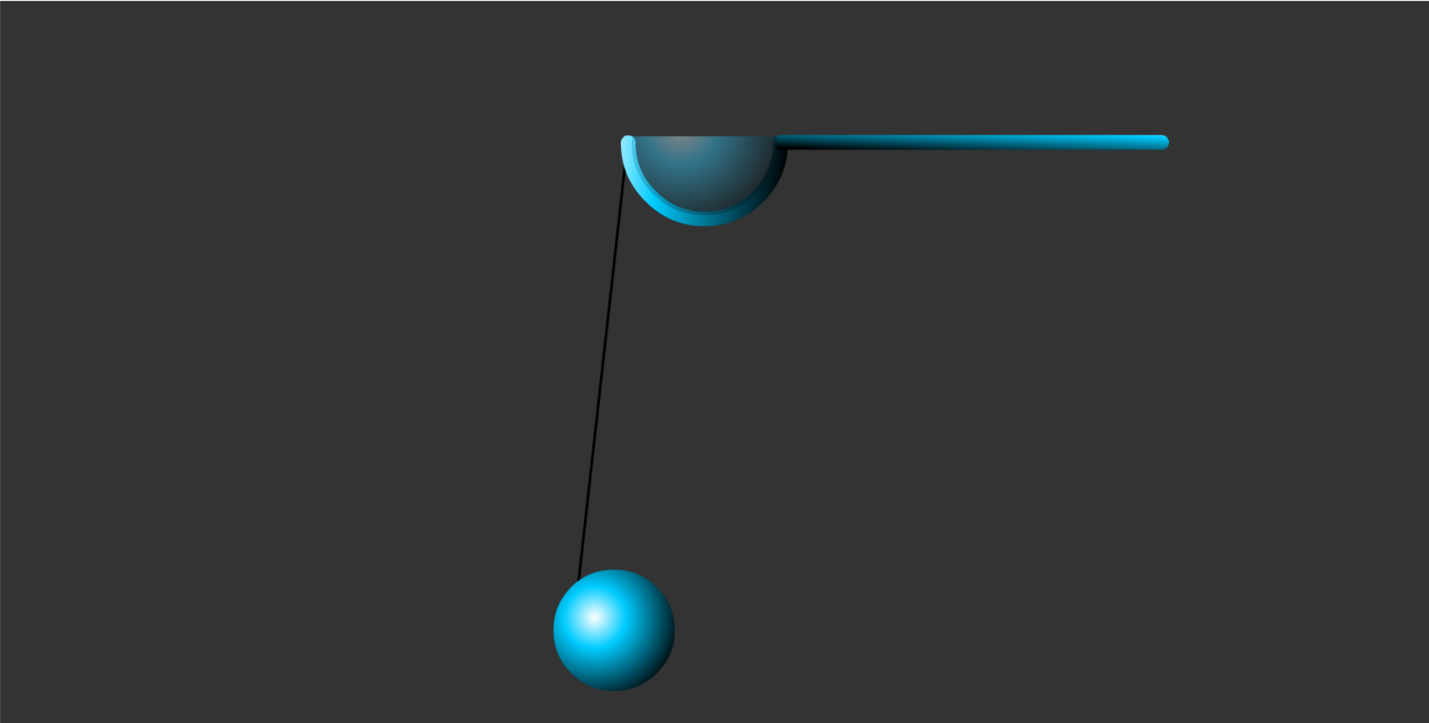
Static friction is used for stationary objects. When applied to an object, it is difficult to move the object at all. Dynamic friction, on the other hand, is used for objects that are already in motion. When applied, the object quickly comes to a stop unless an external force acts on it (acceleration)

1. Find a game that you have seen online or played yourself where physics is a core game mechanic.

You must:

* give a brief description of the game
* briefly explain how physics is used
* explain why physics is important
* include a relevant screenshot of the game

An Adobe Flash-based game, the aptly titled **Bounce Ball Game** is a simple example of physics at play in the context of a game. The game is a digital toy, all you need to do is get the ball into your cup.



Physics were used to simulate the flexibility and tension of the string that the ball is attached to. Additionally, gravity pulls the ball downwards when it is at rest. Without physics, the ball would float due to the lack of gravity and the string would act less like a string in the real world, removing the challenge of the game.

**QUESTION 5 (CharacterController)**

1. According to the video, the built-in CharacterController component does not interact with physics. So, a CharacterController does not react to gravity, and when a CharacterController collides with another object with a rigidbody it does not have any force added to it (like when being hit by a cannonball throws an object across the scene).

What gameplay-related reason(s) can you think of to explain why the CharacterController component was designed this way?

In most games, characters are controlled by player input. Unless physics are a main aspect of the game (and can be readjusted to accommodate this), a player’s gameplay experience might be negatively affected by the physics getting in the way. For example, the character might continue to slide even after there is no more player input, almost as if it is on ice. Therefore, it would be more convenient to have the CharacterController not interact with physics by default as that is more likely to be the option that is used more frequently.

1. To make a character controlled by a CharacterController experience gravity while moving, two options are given in the video. What are these?

One option is to make the character a RigidBody so that Unity knows that the character should be affected by the physics engine, including gravity.

Alternatively, SimpleMove can be applied to the CharacterController, which interacts with gravity by default.

1. Create a Unity scene with a capsule game object with a CharacterController. The script below will make the character jump when the space bar is pressed.

Fill in the missing parts of the code, then copy the whole Character class code here. Make sure it is formatted properly!

Text

Description automatically generated

public class Character : MonoBehaviour

{

CharacterController characterController;

public float speed = 6.0f;

public float jumpSpeed = 8.0f;

public float gravity = 20.0f;

private Vector3 moveDirection = Vector3.zero;

void Start()

{

characterController = GetComponent<CharacterController>();

}

void Update()

{

if (characterController.isGrounded)

{

moveDirection = new Vector3(Input.GetAxis("Horizontal"), 0.0f, Input.GetAxis("Vertical"));

moveDirection \*= speed;

if (Input.GetButton("Jump"))

{

moveDirection.y = jumpSpeed;

}

moveDirection.y -= gravity \* Time.deltaTime;

characterController.Move(moveDirection \* Time.deltaTime);

}

}

}

**QUESTION 6 (Constant Force component)**

1. For the Constant Force component, what is the difference between the Force and Relative Force properties, and between the Torque and Relative Torque properties?

Force moves the object based on the world axis, without taking the orientation of the object into consideration. As its name would imply, Relative force also moves the object, but it moves based on the object’s current orientation, on its local axis. Similarly, Torque spins the object in a direction relative to the world axis, while Relative Torque spins the object in a direction relative to the object’s axis (local).

1. When the cube is in the air with a Force X of 1, it moves, but when it falls to the plane, it doesn’t move.

Why not?

In the air, the cube only has the constant force acting on it. However, the gravity is an opposing force that acts on the cube at the same time, cancelling out the constant force. Therefore, the cube does not move.

1. An object has a constant force applied along its local positive Z axis. What happens to the object’s speed over time?

Explain your answer.

The object’s speed slowly increases. Since the force is applied every frame, it adds up, leading to a speed increase.

**QUESTION 7 (Forces)**

1. Paste your **Player** class code here. Make sure your code is readable, properly formatted, and commented.
2. Paste your **Kick** function code here. Make sure your code is readable, properly formatted, and commented.
3. What does the upwardsModifier argument of AddExplosiveForce do?



**QUESTION 8 (Forces cont.)**

1. What are the FOUR different values for ForceMode? Explain what each value does.
2. Paste your code for firing the sphere here.
3. Paste your code for rotating the beam here.

**QUESTION 9 (Collision event-handling)**

1. Paste your code for **CosmicCube.cs** and **Orb.cs** here. Make sure your code is readable, properly formatted, and commented.

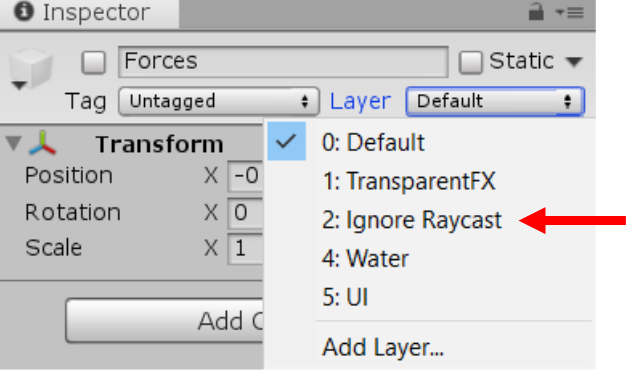
**QUESTION 10 (Raycasting)**

1. Paste your completed code for the CheckLineOfSight() function here.
2. When checking for a hit using a ray, the code below is used:

if (Physics.Raycast(transform.position, vec, out hit, vec.magnitude))

Why is **vec.magnitude** used for the length of the ray?

1. The layers dropdown in the Inspector has an entry for **Ignore Raycast**.



What does this mean? Why do you think this is important? Give an example.