COMSATS University Islamabad, Abbottabad Campus
Department of Computer Science
Midterm Exam

Class: BSE-7A & BSE-7B
Subject: Game Development

Date: 12-Nov-2024
Instructor: Muhammad Ibtisam Gul

Name: Laiba

Reg. # FA21BSE019

**Marks: 30**

**Q 1.** *Understanding (Fundamental Concepts of Game Development).* **(CLO-1, 10 marks)**

a. **Explain** the difference between "game objects" and "components". Provide an **example** of each.

b. **What is** a Prefab? **Describe** two benefits of using Prefabs.

**Q 2.** *Creating (Assets and Scenes for a Game Scenario).* **(CLO-2, 10 marks)**

a. **Create** a list of steps to **implement** player movement using Unity's New Input System. The player should have two input mechanics:

*Thrust: Applies upward movement.*

*Rotation: Rotates the player.*

b. **Outline** the basic structure of a main menu scene, including three main UI components (e.g., Start, Settings, Exit). Also explain how to **implement** user interaction with the UI (e.g. Button Press).

**Q 3.** *Creating (Animations for a Game Scenario).* **(CLO-3, 10 marks)**

a. Consider a game where the player character must navigate around rotating obstacles. Describe each step required to **set up** an obstacle to rotate continuously along the Y-axis and **configure** a Cinemachine Virtual Camera to follow and focus on the player as they move through the scene.

COMSATS UNIVERSITY ISLAMABAD, ABBOTTABAD CAMPUS
Department of Computer Science
Terminal Examination Fall 2024

| | | | | | |
|---|---|---|---|---|---|
| **Subject & Class:** | *Game Development - BSE-7A/7B* | **Instructor:** | *Dr. M. Ibtisam Gul* | **Date:** | *24-Jan-25* |
| **Time Allowed:** | *180 Minutes* | **Marks:** | *60* | | |

**INSTRUCTIONS: Do 2 out of 3 questions from each section.**

### CLO-1: Characterize the fundamental concepts of game development

Q.1. Explain the purpose of MonoBehaviour and ScriptableObject in Unity scripts. Provide examples of when to use MonoBehaviour and ScriptableObject.

Q.2. Describe how to create a basic Unity UI button and link it to a function that loads a new scene. Include details about how to set up the button, attach the script to load the new scene.

Q.3. Explain the purpose of physics materials in Unity. Discuss how to create a realistic bouncy ball using a Rigidbody, Collider, and Physics Material, and specify the properties to adjust for desired bounce behavior.

### CLO-2: Create different assets and scenes for a game scenario

Q.4. Write a script to play a sound effect when the player collects a specific item in the game. The script should destroy the item upon collection and trigger the sound effect.

Q.5. Use Unity's Terrain tools to create a basic landscape with hills, textures, and trees. How can you edit and customize a basic ProBuilder Cube to fit into this landscape.

Q.6. Implement a basic first-person controller for forward and backward movement using Unity's New Input System. Additionally, create a follow camera that smoothly tracks the player's movement.

### CLO-3: Create animations for a game scenario

Q.7. Describe the steps to create a basic animation clip for a to-and-fro movement of an object (e.g., a moving obstacle) and attach it to a GameObject using Unity's Animator.

Q.8. Discuss how to apply root motion to a humanoid character animation in Unity. Explain why root motion is necessary in certain scenarios.

Q.9. Create a system where a character switches to Ragdoll Physics upon death and reverts to its original state upon respawn. Describe how to enable and disable Ragdoll components programmatically.

### CLO-4: Illustrate the concept of C# scripting and AI for gaming

Q.10. Implement an enemy AI system that follows the player when detected and attacks after reaching close range. Describe how to use raycasting and distance calculation to detect the player.

Q.11. Implement an AI-controlled character that follows the player character while avoiding obstacles on a plane. Use Unity's NavMesh system for pathfinding and explain how to set up navigation areas and obstacles in the game world.

Q.12. Investigate the working of the UpdateWheelTransform function in the VehicleController script. Why is it important to call GetWorldPose? Explain how it allows proper wheel position and rotation alignment with the vehicle's movement.

```
void UpdateWheelTransform(WheelCollider wheelCollider, Transform wheelTransform)
{
    Vector3 pos;
    Quaternion rot;
    wheelCollider.GetWorldPose(out pos, out rot);
    wheelTransform.position = pos;
    wheelTransform.rotation = rot;
}
```