

Course Code	Course Title	Credits	Lectures /Week
USCS5042	Game Programming	2	3
<p>About the Course: This course provides an in-depth exploration of 3D game development and graphics, covering topics such as vectors, transformations, 3D modeling, rendering, physics-based simulation, and game engine architecture. Students will gain hands-on experience using industry-standard tools and technologies, including DirectX, Unity, and Python-Pygame. Through a combination of theoretical study, practical exercises, and project-based learning, students will develop the skills necessary to create immersive 3D games and interactive virtual worlds.</p>			
<p>Course Objectives:</p> <ul style="list-style-type: none"> • Understand the fundamentals of vectors, transformations, and 3D graphics. • Develop proficiency in using industry-standard tools and technologies for 3D game development. • Implement advanced graphics techniques, including lighting, shading, and texturing, to create visually appealing game environments. • Apply principles of game design and create engaging and immersive gaming experiences. 			
<p>Learning Outcomes: After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> • Apply vector manipulation techniques and transformations to create and manipulate objects in 3D space. • Utilize industry-standard tools and technologies such as Unity and DirectX for 3D game development. • Implement advanced graphics techniques, including lighting, shading, and texturing, to create visually stunning game environments. • Design and develop games that incorporate principles of game design to create engaging and immersive experiences. • Deploy and showcase 3D games on various platforms, demonstrating proficiency in game development. 			
Unit	Topics	No of Lectures	
I	<p>Introduction to Vectors: Vectors: Vector Manipulation, multiplying a Vector by a Scalar, Vector Addition and Subtraction, Position Vectors, Unit Vectors, Cartesian Vectors, Vector Multiplication, Scalar Product, Example of the Dot Product, The Dot Product in Lighting Calculations, The Dot Product in Back-Face Detection, The Vector Product, The Right-Hand Rule, deriving a Unit Normal Vector for a Triangle Areas, Calculating 2D Areas</p> <p>Transformations: 2D Transformations, Matrices, Homogeneous Coordinates, 3D Transformations, Change of Axes, Direction Cosines, rotating a Point about an Arbitrary Axis, Transforming Vectors, Determinants, Perspective Projection, Interpolation</p> <p>3D Graphics for Game Programming: 3D Transformations, Quaternions,</p>	15	

	3D Modeling and Rendering, Ray Tracing, Shader Models, Lighting, Color, Texturing, Camera and Projections, Culling and Clipping, Character Animation, Physics-based Simulation, Scene Graphs.	
II	<p>Game Engines and Design: Game engine architecture, Engine support systems, Resources and File systems, Game loop and real-time simulation, Human Interface devices, Collision and rigid body dynamics, Game profiling. Introduction to Dx Studio, Introduction to Unity, Application layer, Game logic, Game views, managing memory, controlling the main loop, loading and caching game data, User Interface management, Game event management.</p> <p>Introduction To DirectX 11: COM, Textures and Resources Formats, The swap chain and Page flipping, Depth Buffering, Texture Resource Views, Multisampling Theory and MS in Direct3D, Feature Levels</p> <p>Game Platform and Development: 2D and 3D Game development using Python-Pygame, ModernGL, Numpy, PyGLM,Ursina (OpenGL)</p>	15
III	<p>Unity Development Environment: IDE Basics, Unity Concepts, Sprites, Game Loops and Functions, Game Design Strategies.</p> <p>Simple Movement, Operations and Object Oriented Concepts: Simple Rotation and Scaling, Rigidbody Components, Unity Colliders, Physics Materials, Scripting Collision Events, Primitive Data and Math, Decisions and Flow Control, Loops and Arrays, Game Design Strategies, Exceptions and Debugging, Defining Classes, Functions, Organizing and Managing Game Objects,</p> <p>Advanced Programming: Virtual Worlds, Scrolling Games, Animation, Sound Effects, Advanced Game Physics, Multiple Scenes, Artificial Intelligence, User Interfaces, Game Art, Publishing Games.</p>	15
<p>Textbook(s):</p> <ol style="list-style-type: none"> 1. Mathematics for Computer Graphics, John Vince, Springer-Verlag London, 5th Edition,2017 2. Mathematics for 3D Game Programming and Computer Graphic, Eric Lengyel, Delmar 3. Introduction To 3D Game Programming With Directx® 11,Frank D Luna, Mercury Learning And Information,2012. <p>Additional Reference(s):</p> <ol style="list-style-type: none"> 1. Computer Graphics, C Version, Donald Hern and Pauline Baker, Pearson Education, 2nd Edition, 1997 2. HLSL Development Cookbook, Doron Feinstein, PACKT Publishing,2013 3. https://docs.unity3d.com/Manual/index.html 		