GANPAT UNIVERSITY

FACULTY OF ENGINEERING & TECHNOLOGY

Programme	Bachelor of Technology	Branch/Spec.	Computer Engineering / Information Technology
Semester	V	Version	2.0.0.0
Effective from Academic Year	2020-21	Effective for the batch Admitted in	July 2018
Subject code	2CEIT5PE2	Subject Name	Computer Graphics & Visualization

Teaching scheme						Examination scheme (Marks)			
(Per week)	Lecture(DT)		Practical(Lab.)		Total		CE	SEE	Total
	L	TU	P	TW					
Credit	3	0	1	-	4	Theory	40	60	100
Hours	3	0	2	-	5	Practical	30	20	50

Pre-requisites:

Knowledge of C Programming language, Basic Linear Algebra, Basic data structures and algorithms

Objectives of the course:

1. Understand about fundamentals of Graphics to enable them to design animated scenes for virtual object creations.
2. Understand the 2D & 3D graphics and their transformations.
3. Understand illumination and color models.
4. Understand clipping techniques.

Theory Syllabus

Unit	Content	Hrs
1	**Introduction:** History of Computer Graphics, Applications, Animation, Rendering, Relation to Computer Vision and Image Processing	02
2	**Introduction To Opengl:** Opengl Architecture, Primitives and Attributes, Simple Modeling and Rendering of Two- and Three-Dimensional Geometric Objects, Indexed and RGB Color Models, Frame Buffer, Double Buffering, GLUT, Interaction, Events and Callbacks, Picking.	04
3	**Raster Graphics & Clipping:** Point, Line, Circle and Ellipse as Primitives, Fill Area Primitives, Windowing, Line Clipping (Cohen and Sutherland, Cyrus-Beck), Polygon Clipping, 3d Clipping, Introduction to Anti-Aliasing Technique	05
4	**2D & 3D Geometric Transformations:** Basic Transformation, Matrix Representations and Homogenous Coordinates, Composite Transformations, Affine Transformations, Matrix Stacks and Model View Matrix In Opengl.	04
5	**2D & 3D Viewing:** Viewing Pipeline and Co-Ordinate System, Viewing Transformations, Classical 3D Viewing, Parallel and Perspective Projective Transformations	02
6	**3D Object Representation and Visualization:** Curves and Surfaces: Cubic Splines, Bezier Curves, B-Splines, Tensor Product Surfaces, Surface of Revolution Sweep Surfaces, Fractal Curves and Surfaces, Hidden Line/Surface Removal Methods, Visibility- Z-Buffer, BSP Trees, Open-GL Culling, Hidden-Surface Algorithms, Visualization, Interpolation, Modelling Techniques, Trees, Scene Graphs, Wireframe, Surface and Solid Modelling, Surface Area and Volume Estimation For 3D Tessellation	09
7	**Lighting & Shading:** Light Sources, Basic Illumination Models: Ambient Light, Diffuse Reflection, Specular Reflection	08

	and Phong Model, Intensity Attenuation, Color Models, Transparency and Shadows, Gouraud and Phong Shading for Polygons, Programmable Shaders: Opengl Shading Language, Fragment Shaders, Cub and Bump Maps	
8	**Discrete Techniques:** Texture Mapping, Compositing, Textures in Opengl; Ray Tracing- Recursive Ray Tracer, Ray-Sphere Intersection	05
9	**Introduction of CUDA Programming:** Different Generations of Gpus, GPU Architecture Overview, CUDA Programming Model, Memory Models, CUDA Hardware Interface on The GPU, CUDA Programming Examples	03
10	**Introduction of Animation:** Principles of Animation, Overview of Various Animation Techniques, Storyboards for Animation, Key-Frame System, Tweening and Morphing	03

Practical Content	
Experiments/Practical/Simulations would be carried out based on syllabus	
Text Books	
1	"Mathematics for 3D Game Programming and Computer Graphics" by Eric Lengyel, Course Technology PTR Cengage Learning.
2	Donald Hearn and Pauline Baker, Computer Graphics with OpenGL, Prentice Hall
Reference Books	
1	Peter Shirley and Steve Marschner, Computer Graphics, A. K. Peters.
2	F. S. Hill Jr. and S. M. Kelley, Computer Graphics using OpenGL, Prentice Hall.
3	"Geometric Modeling and Mesh Generation from Scanned Images by Jessica Zhang", Taylor and Francis Group.
4	"Curves and Surfaces For CAGD" by Gerald Farin, Morgan Kauffman Publishers.
5	"Polygon Mesh Processing" by Botsch, Kobbelt, Pauly, Alliez and Levy, A K Peters Ltd
6	Tomas Akenine-Mller and Eric Haines Naty Hoffman, Real-Time Rendering, A.K. Peters.
7	NVidia CUDA Repository, URL: http://developer.nvidia.com/category/zone/cudazone
8	Donald Hearn and M Pauline Baker, "Computer Graphics C Version", Pearson Education
9	"Geometric Modeling and Mesh Generation from Scanned Images by Jessica Zhang", Taylor and Francis Group.
10	M. de Berg, M. Van Kreveld, M. Overmars, and O. Schwarzkopf, Computational Geometry: Algorithms and Applications, Springer.
ICT/MOOCs Reference	
1	https://nptel.ac.in/courses/106106090/

Course Outcomes:

After successful completion of this course, student will be able to

1. Implement the concept of 2D and 3D transformations, projection and viewing.
2. Gain detailed knowledge of the graphics pipeline.
3. Implement the concept of shading and texture mapping algorithms.
4. Get broad knowledge of 3D modelling and rendering techniques.
5. Understand, design and implement scene graphs for games.
6. Develop practical skills in graphics programming.