3/4 B.Tech. FIRST SEMESTER

CS5T3 **COMPUTER GRAPHICS** **Credits: 4**

(Common to CSE & IT)

Required

Lecture: 4 periods/week **Internal assessment: 30 marks**

Tutorial: 1 period /week **Semester end examination: 70 marks**

Course context and Overview: Computer Graphics I is a study of the hardware and software principles of interactive raster graphics. Topics include an introduction to the basic concepts, 2-D and 3-D modeling and transformations, viewing transformations, projections, rendering techniques, graphical software packages and graphics systems. Students will use a standard computer graphics API to reinforce concepts and study fundamental computer graphics algorithms.

Prerequisites: Vector, Vector Operations, Vector Spaces, Matrices, Basic Linear Algebra such as solving a system of Linear Equations, Polynomials, Elementary signal processing(Fourier transform and filtering)

Objectives:

1. Students should learn the basics of application programming interface (API) implementation based on graphics pipeline approach.
2. Student should study and understand graphics through OpenGL.
3. Students should learn the basic input devices and interaction of computer graphics.
4. Student will have a thorough understanding of the fundamentals of 2D and 3D computer graphics.
5. Understand the modern graphical hardware and rendering pipeline.
6. Students should learn to use mathematical transformations and vector techniques in the production of computer graphics.
7. Students should gain familiarity of clipping algorithms, rasterization techniques.

Learning Outcomes:

Upon successful completion of the course, students will

Ability to:

1. Understand graphics applications, architectures and openGL program structure.
2. Understand lighting and shading models and hidden surface removal methods.
3. Apply basic transformations on objects.
4. Apply line and polygon clipping algorithms.
5. Illustrate different projections.
6. Design interactive programs using openGL.

UNIT I

Introduction:

Applications of computer graphics; A graphics system; Images: Physical and synthetic; Imaging systems; The synthetic camera model; The programmer's interface; Graphics architectures; Programmable pipelines; Performance characteristics.

Graphics Programming: The Sierpinski gasket; Programming two- dimensional applications.

UNIT II
The OpenGL:

The OpenGL API; Primitives and attributes; Color; Viewing; Control functions; The Gasket program; Polygons and recursion; The three-dimensional gasket; Plotting implicit functions.

UNIT III
Input and Interaction:

Interaction; Input devices; Clients and servers; Display lists; Display lists and modeling; Programming event-driven input; Menus; Picking; A simple CAD program; Building interactive models; Animating interactive programs; Design of interactive programs; Logic operations.

UNIT IV
Geometric Objects and Transformations – 1:

Scalars, points, and vectors; Three-dimensional primitives; Coordinate systems and frames; Modeling a colored cube; Affine transformations; Rotation, translation and scaling.

UNIT V
Geometric Objects and Transformations – 2:

Transformations in homogeneous coordinates; Concatenation of transformations; OpenGL transformation matrices; Interfaces to three-dimensional applications; Quaternions.

UNIT VI
Viewing:

Classical and computer viewing; Viewing with a computer; Positioning of the camera; Simple projections; Projections in OpenGL; Hidden-surface removal; Interactive mesh displays; Parallel-projection matrices; Perspective-projection matrices; Projections and shadows.

UNIT VII
Lighting and Shading:

Light and matter; Light sources; The Phong lighting model; Computation of vectors; Polygonal shading; Approximation of a sphere by recursive subdivisions; Light sources in OpenGL; Specification of materials in OpenGL; Shading of the sphere model; Global illumination.

UNITVIII
Implementation

Basic implementation strategies; The major tasks; Clipping; Line-segment clipping; Polygon clipping; Clipping of other primitives; Clipping in three dimensions; Rasterization; Bresenham's algorithm; Polygon rasterization; Hidden-surface removal; Antialiasing; Display considerations.

Learning Resources

Text Books:

1. Interactive Computer Graphics A Top-Down Approach with OpenGL, Edward Angel, Pearson, 5th Edition, 2009. (Chapters 1, 2, 3, 4, 5, 6, 7)
2. Computer Graphics Through OpenGL: From Theory to Experiments, Sumantha Guha, Chapman and Hall/CRC, 2011 (For OpenGL and related examples).

Reference Books:

1. Computer Graphics with OpenGL, Hearn & Baker, 3rd Edition, Pearson Education.
2. Computer Graphics Using OpenGL, F.S. Hill,Jr, and M. Kelley, Jr., Pearson/PHI, 3rd Edition, 2009.