**CS2401 COMPUTER GRAPHICS L T P C 3 0 0 3**

# AIM

# To have an understanding of foundations of 2D and 3D Concepts, Graphics programming, rendering, fractals

**OBJECTIVES**

* To understand the 2D and 3D concepts
* To implement the algorithms
* To be able to do graphics programming

**UNIT I 2D PRIMITIVES 9**

output primitives – Line, Circle and Ellipse drawing algorithms - Attributes of output primitives – Two dimensional Geometric transformation - Two dimensional viewing – Line, Polygon, Curve and Text clipping algorithms

**UNIT II 3D CONCEPTS 9**

Parallel and Perspective projections - Three dimensional object representation – Polygons, Curved lines, Splines, Quadric Surfaces,- Visualization of data sets - 3D transformations – Viewing -Visible surface identification.

**UNIT III GRAPHICS PROGRAMMING 9**

Color Models – RGB, YIQ, CMY, HSV – Animations – General Computer Animation, Raster, Key frame - Graphics programming using OPENGL – Basic graphics primitives – Drawing three dimensional objects - Drawing three dimensional scenes

**UNIT IV RENDERING 9**

Introduction to Shading models – Flat and Smooth shading – Adding texture to faces – Adding shadows of objects – Building a camera in a program – Creating shaded objects – Rendering texture – Drawing Shadows.

**UNIT V FRACTALS 9**

Fractals and Self similarity – Peano curves – Creating image by iterated functions – Mandelbrot sets – Julia Sets – Random Fractals – Overview of Ray Tracing – Intersecting rays with other primitives – Adding Surface texture – Reflections and Transparency – Boolean operations on Objects.

**TOTAL = 45 PERIODS**

**TEXT BOOKS:**

1. Donald Hearn, Pauline Baker, Computer Graphics - C Version, second edition,PearsonEducation,2004.

2. F.S. Hill, Computer Graphics using OPENGL, Second edition, Pearson Education,2003.

**REFERENCES:**

1. James D. Foley, Andries Van Dam, Steven K. Feiner, John F. Hughes, Computer

Graphics- Principles and practice, Second Edition in C, Pearson Education, 2007.

**CS2405 COMPUTER GRAPHICS LABORATORY L T P C 0 0 3 2**

1. Implementation of Bresenhams Algorithm – Line, Circle, Ellipse.

2. Implementation of Line, Circle and ellipse Attributes

3. Two Dimensional transformations - Translation, Rotation, Scaling, Reflection, Shear.

4. Composite 2D Transformations

5. Cohen Sutherland 2D line clipping and Windowing

6. Sutherland – Hodgeman Polygon clipping Algorithm

7. Three dimensional transformations - Translation, Rotation, Scaling

8. Composite 3D transformations

9. Drawing three dimensional objects and Scenes

10. Generating Fractal images

**TOTAL: 45 PERIODS**

**LIST OF EQUIPMENTS:**

1) Turbo C

2) Visual C++ with OPENGL

3) Any 3D animation software like 3DSMAX, Maya, Blender