

<b>Devi Ahilya University, Indore, India Institute of Engineering &amp; Technology</b>			<b>IV Year B.E. (Information Technology (Full Time))</b>			
<b>Subject Code &amp; Name</b>	<b>Instructions Hours per Week</b>			<b>Credits</b>		
<b>7ITRC2 Computer Graphics</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>L</b>	<b>T</b>	<b>P</b>
<b>Duration of Theory Paper: 3 Hours</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>1</b>

### Learning Objective:

The course aims to:

- Introduce the core concepts and algorithms used in computer graphics.
- Develop proficiency in the use of graphics programming languages and tools.
- Foster an understanding of the mathematical foundations underlying graphic transformations and projections.
- Enable students to design and implement graphics systems and applications.
- Encourage creativity and innovation in the design of graphical user interfaces and visualizations.

### Pre requisites:

Programming language: C++/JAVA.

## COURSE CONTENTS

### UNIT-I

Introduction: Application of Computer Graphics, Raster Graphics Fundamentals: Scan conversion, Pixel, Frame Buffer. Graphics Primitives; Line algorithms Circle algorithms, Ellipse, Character generation, Polygon Representation, inside test, Polygon filling algorithms, Antialiasing.

### UNIT-II

Display devices: Random scan and Raster scan monitors, Colors CRT monitor, Plasma Panel; Hard Copy devices: Printers and Plotters; Input devices: Joysticks, Mouse, Digitizer, Scanner, and Camera; Input Techniques;

### UNIT-III

Windowing and clipping: 2D Transformation, Raster method of Transformation, Window, View port, Viewing, Window to View port Transformation, Line clipping algorithms, Polygon clipping algorithms.

### UNIT-IV

Three Dimensions: 3D Modeling techniques, 3D Display Techniques, 3D Transformation, Viewing Parameters, Hidden Surface and back face removal algorithms. 3D Curves & Surfaces: Bezier, B spline.

## **UNIT-V**

Shading and Color Models: Diffuse illumination, Point source illumination, Reflection, Refraction, Transparency, Shadows, Polygon rendering algorithms, Dithering, Half toning, Color Models and applications.

### **BOOKS RECOMMENDED:**

- [1] Hearn Donald and Baker M.Pauling, Computer Graphics, 2/e, Prentice Hall of India.
- [2] Hearn Donald and Baker M.Pauling, Computer Graphics with OpenGL, 3/e, Prentice Hall,
- [3] David F. Rogers, Procedural Element of computer Graphics, McGraw Hill International.
- [4] William M. Newman Robert F. Sproull, Principles of Interactive Computer Graphics, McGraw Hill.
- [5] J.D. Foley, A. van Dam, S.K. Feiner, J.F. Hughes, and R.L. Philips, Introduction to ComputerGraphics, Addison-Wesley, 1994.
- [6] Zhigang Xiang and Roy Plastock, Computer Graphics, Tata McGraw Hill Publications.

### **List of Practical Assignments:**

1. Implement various line drawing algorithm & compare it on the basis of performance and prepare the table.
2. Implement various Circle drawing algorithm & compare it on the basis of performance and prepare the table.
3. Implement Cohen Sutherland and Cyrus Back line clipping algorithms and display the clipped portion of line as demonstration.
4. Implement Cohen Hoghman polygon clipping algorithm to clip a polygon against a rectangular boundary and combine Cohen Hoghman polygon clipping algorithm with Cyrus Back line clipping algorithms to clip a polygon against another polygon and display the clipped portion of polygon.
5. Implement the following 2D transformation using matrix multiplication: Translation Rotation Scaling Shearing Reflection
6. Implement parallel (oblique and axonometric) and perspective projection and display following objects using projections: Pyramid and Rectangles
7. Implement the following 3D transformation using matrix multiplication. And display using perspective projection: Translation Rotation Scaling Shearing Reflection
8. Implement rotation of an object about any arbitrary axis.
9. Implement reflection of an object about any arbitrary plane.
10. Generate 2D curve and surfaces using implementation of Bezier curve.
11. Implement the following visible surface detection algorithm and compare them on the basis of complexities and execution time. Z-buffer Scan-line Depth sorting Octree Ray Casting.
12. Implement the following rendering algorithm and compare them on the basis of complexities and execution time. Flat Shading Gouraud shading Phong shading Ray tracing.

### **Program Outcomes (POs):**

#### **PO1: Engineering Principles and Problem-Solving in Computer Graphics**

Demonstrate a solid understanding of engineering principles and apply them to solve complex problems in computer graphics and related fields.

#### **PO2: Proficiency in Graphics Programming and Tools**

Description: Exhibit proficiency in programming and the use of specialized graphics software and tools.

#### **PO3: Critical Thinking and Innovation in Graphical Solutions**

Description: Apply critical thinking and problem-solving skills to develop innovative graphical solutions.

#### **PO4: Teamwork and Communication**

Work effectively in multidisciplinary teams and communicate complex technical concepts clearly and effectively.

#### **PO5: Lifelong Learning and Adaptability in Computer Graphics**

Engage in lifelong learning to stay abreast of advancements in computer graphics technologies and practices.

#### **PO6: Ethical and Social Responsibility in Computer Graphics**

Understand the ethical, social, and professional responsibilities related to the development and use of computer graphics.

#### **PO7: Research and Experimental Techniques in Computer Graphics**

Conduct research and apply experimental techniques to explore new frontiers in computer graphics.

#### **PO8: Interdisciplinary Integration and Technological Development**

Integrate knowledge from computer graphics with other engineering domains to contribute to the development of cutting-edge technologies and applications.

### **Course Outcomes (COs)**

#### **CO1: Fundamental Concepts of Computer Graphics**

Understand and explain the fundamental concepts of computer graphics, including graphics hardware, software, and applications.

#### **CO2: Mathematical Modeling and Rendering**

Apply mathematical principles and algorithms to model and render 2D and 3D graphics.

#### **CO3: Graphics Programming and Libraries**

Develop and implement graphics programs using programming languages and libraries such as OpenGL or DirectX.

#### **CO4: Geometric Transformations**

Perform geometric transformations including translation, scaling, rotation, and shearing on graphical objects.

#### **CO5: Complex Scene Management**

Create and manage complex scenes with multiple objects, lighting, shading, and texture mapping.

#### **CO6: Rendering Techniques for Realistic Images**

Utilize various rendering techniques, including rasterization and ray tracing, to produce realistic images.

#### **CO7: GUI Design and Interactive Graphics Applications**

Design and evaluate graphical user interfaces (GUIs) and interactive graphics applications.

#### **CO8: Real-World Problem-Solving with Computer Graphics**

Solve real-world problems using computer graphics techniques and tools.

## Matrix for CO-PO Relationship

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	-	-	1	1	-	-
CO2	3	3	2	-	-	-	1	1
CO3	3	2	3	-	-	-	-	1
CO4	3	-	2	-	-	-	-	-
CO5	3	2	2	-	-	-	-	-
CO6	3	2	2	-	-	-	1	-
CO7	3	2	2	1	-	1	-	-
CO8	3	2	2	-	-	-	1	1

In this matrix:

- - indicates no contribution.
- 1 indicates a relevant and small significance.
- 2 indicates a medium or moderate contribution.
- 3 indicates a strong contribution.

This matrix clearly shows the contribution levels of each course outcome (CO) to the respective program outcomes (PO).

## CO -PO Relationship

Course Outcome No.	Course Outcome Details	Supported Program Outcomes
CO1	Understand and explain the fundamental concepts of computer graphics, including graphics hardware, software, and applications.	PO1, PO2, PO5, PO6
CO2	Apply mathematical principles and algorithms to model and render 2D and 3D graphics.	PO1, PO2, PO3, PO7, PO8
CO3	Develop and implement graphics programs using programming languages and libraries such as OpenGL or DirectX.	PO1, PO2, PO3, PO8
CO4	Perform geometric transformations including translation, scaling, rotation, and shearing on graphical objects	PO1, PO3
CO5	Create and manage complex scenes with multiple objects, lighting, shading, and texture mapping.	PO1, PO2, PO3
CO6	Utilize various rendering techniques, including rasterization and ray tracing, to produce realistic images.	PO1, PO2, PO3, PO7
CO7	Design and evaluate graphical user interfaces (GUIs) and interactive graphics applications.	PO1, PO2, PO3, PO4, PO6
CO8	Solve real-world problems using computer graphics techniques and tools.	PO1, PO2, PO3, PO7, PO8