DISCIPLINE SPECIFIC Elective - (DSE-4b) : Foundations of Computer Graphics

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Semester 6

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
DSE 04b Foundations of Computer Graphics	**4**	**3**	**0**	**1**	Pass in Class XII	**Programming in C/C++, Basic Matrices and Differential Calculus**

Learning Objectives

This course lays the foundation of Computer Graphics by introducing the fundamental concepts of modelling, rendering and interaction. The course focuses on rasterization technique for discussing the drawing and rendering of primitives. The course gives both the theoretical and practical knowledge to design, use, and understand computer graphics systems.

Learning outcomes

On successful completion of the course, students will be able to:

1. Describe Standard raster and vector scan devices as well as Graphical Input and output devices
2. Present the different stages of the traditional graphics pipeline in detail.
3. Implement algorithms for drawing basic primitives such as line and circle.
4. Implement algorithms for line clipping and polygon clipping.
5. Explain how models can be translated, scaled, rotated, and sheared by transformation matrices including the use of homogeneous coordinates and concatenated transformations
6. Understand and explain the basic principles of visible surface determination while rendering a scene.
7. Describe the process to compute the intensity at a particular point in the scene and use the same to shade the entire surface.
8. Understand the basics of simple computer animation.

SYLLABUS OF DSE 04b

Unit 1 **(5 hours)**

Introduction: Introduction to Graphics systems, Basic elements of Computer graphics, Applications of computer graphics, Architecture of Raster and Vector scan display devices, Color Lookup Table, Display devices (Cathode Ray Tube (CRT) , Colored CRTs, Direct View Storage tube(DVST), Plasma Panel, LCD, LED, Emissive and Non-emissive displays), Input devices.

Unit 2 **(8 hours)**

Drawing and clipping primitives: Raster scan line (Digital Differential Analyzer (DDA) and Bresenhams) and circle drawing algorithms, line clipping using Cohen and Sutherland Line clipping algorithm and polygon clipping using Sutherland and Hodgeman Polygon clipping algorithm.

Unit 3 **(15 hours)**

Transformation and Viewing: Basic 2D transformations (Translation, Rotation, scaling, reflection and shearing), Homogenous coordinates, composite transformation, 3D Geometric Transformations, Viewing Transformations (Projections- Parallel and Perspective), Vanishing points.

Unit 4 **(5 hours)**

Geometric Modeling: Polygon Mesh Representation, Cubic Polynomial curves (Hermite).

Unit 5 **(8 hours)**

Visible Surface determination/Hidden Surface removal and Surface Rendering: Need for hidden surface removal, Z-buffer algorithm and area subdivision algorithm for visible surface determination. Phong Illumination model, Phong and Gouraud shading models, Halftoning and Dithering.

Unit 6 **(4 hours)**

Basics of Computer Animation: Storyboard layout, keyframe systems, simulating motion, morphing.

Essential/recommended readings

1. Hearn, D & Baker, M.P. Computer Graphics. 2nd edition. Prentice Hall of India, 2009.
2. Foley, J. D., Dam, A.V, Feiner, S. K., & Hughes, J. F. . Computer Graphics: Principles and Practice in C. 2nd edition. Pearson Education, 2002.
3. Rogers, D. F. . Mathematical Elements for Computer Graphics. 2nd edition. McGraw Hill. 2017.

Additional References:

1. Bhattacharya, S. Computer Graphics. Oxford University Press, 2018.
2. Marschner, S., & Shirley, P. Fundamentals of Computer Graphics. 5th edition. CRC Press, 2021.

Online references/material:

Suggested Practical List (If any): (30 Hours)

1. Write a program to implement Bresenham's line drawing algorithm.
2. Write a program to implement mid-point circle drawing algorithm.
3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm.
4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.
5. Write a program to apply various 2D transformations on a 2D object (use homogenous Coordinates).
6. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.
7. Write a program to draw the Hermite curve.