

**Name of the Faculty:** Ms. Java  
**Discipline:** B.Tech CSE  
**Semester:** 7th Sem  
**Subject:** Computer Graphics and Animation ( CSE- 403N)  
**Work Load (Lecture/Practical) per week (in hours):** Lectures- 04 , Practicals- 02

Week	Theory		Practical	
	Lecture day	Topic (including assignment/test)	Practical day	Topic
1 <sup>st</sup>	1 <sup>st</sup>	Computer Graphics	1 <sup>st</sup>	Write a program to implement DDA line drawing algorithm
	2 <sup>nd</sup>	Computer Graphics applications		
	3 <sup>rd</sup>	Display Devices		
	4 <sup>th</sup>	Point & Positioning Devices		
2 <sup>nd</sup>	5 <sup>th</sup>	Plotting Techniques for point and Line	2 <sup>nd</sup>	Write a program to implement Bresenham's line drawing algorithm
	6 <sup>th</sup>	Line drawing algorithms		
	7 <sup>th</sup>	DDA Algorithm		
	8 <sup>th</sup>	Bresenham's Circle drawing algorithms		
3 <sup>rd</sup>	9 <sup>th</sup>	Bresenham's Circle drawing algorithms	3 <sup>rd</sup>	Implement the Bresenham's circle drawing algorithm.
	10 <sup>th</sup>	Filled area algorithms		
	11 <sup>th</sup>	Scan line		
	12 <sup>th</sup>	Polygon filling algorithms		
4 <sup>th</sup>	13 <sup>th</sup>	Boundary filled algorithms	4 <sup>th</sup>	Write a program to draw a decagon whose all vertices are connected with every other vertex using lines
	14 <sup>th</sup>	Revision		
	15 <sup>th</sup>	Test		
	16 <sup>th</sup>	Window to view port transformation		
5 <sup>th</sup>	17 <sup>th</sup>	Window to view port mapping,	5 <sup>th</sup>	Write a program to move an object using the concepts of 2-D transformations.
	18 <sup>th</sup>	Two Dimensional transformation: translation		
	19 <sup>th</sup>	Two Dimensional transformation: scaling		
	20 <sup>th</sup>	Two Dimensional transformation : rotation		
6 <sup>th</sup>	21 <sup>st</sup>	two dimensional Transformation : Reflection	6 <sup>th</sup>	Write a program to implement the midpoint circle drawing algorithmany Object Oriented Programming Language like Python, C++,Java.
	22 <sup>nd</sup>	Two Dimensional transformation : Shear		
	23 <sup>rd</sup>	Homogeneous Coordinate system.		
	24 <sup>th</sup>	3-D transformation: Rotation		
	25 <sup>th</sup>	3-D transformation: Shear		

7 <sup>th</sup>	26 <sup>th</sup>	3-D transformation: translation	7 <sup>th</sup>	Implement the line clipping algorithm using any Object Oriented Programming Language like Python, C++, Java.
	27 <sup>th</sup>	Numerical Problems of transformation viewing pipeline		
	28 <sup>th</sup>	Revision		
8 <sup>th</sup>	29 <sup>th</sup>	Test	8 <sup>th</sup>	Implement boundary fill algorithm using any Object Oriented Programming Language like Python, C++, Java
	30 <sup>th</sup>	Clipping: Point clipping algorithm		
	31 <sup>st</sup>	4-bit code algorithm		
	32 <sup>nd</sup>	Cohen-Sutherland Line clipping algorithms		
9 <sup>th</sup>	33 <sup>rd</sup>	Liang-Barsky line clipping algorithms.	9 <sup>th</sup>	Implement the depth buffer algorithm using any Object oriented language like Python, C++,Java
	34 <sup>th</sup>	Polygon clipping		
	35 <sup>th</sup>	Sutherland-Hodgeman Polygon clipping algorithm		
	36 <sup>th</sup>	Curve clipping		
10 <sup>th</sup>	37 <sup>th</sup>	Text clipping	10 <sup>th</sup>	Perform the Polygon Clipping Algorithm using any Object oriented language like Python, C++,Java
	38 <sup>th</sup>	Projection: Parallel		
	39 <sup>th</sup>	Projection Perspective		
	40 <sup>th</sup>	Vanishing Points		
11 <sup>th</sup>	41 <sup>st</sup>	Revision	11 <sup>th</sup>	Draw a Rectangle using Bresenham's and DDA Algorithm using any Object oriented language like Python, C++,Java.
	42 <sup>nd</sup>	Test		
	43 <sup>rd</sup>	Representation of 3-D Curves and Surfaces		
	44 <sup>th</sup>	Interpolation		
12 <sup>th</sup>	45 <sup>th</sup>	Interpolation		
	46 <sup>th</sup>	approximation alpines		
	47 <sup>th</sup>	parametric conditions		
	48 <sup>th</sup>	Geometric continuity conditions		
13 <sup>th</sup>	49 <sup>th</sup>	Beizer curves and surfaces		
	50 <sup>th</sup>	properties of beizer curves		
	51 <sup>st</sup>	beizer surfaces.		
	52 <sup>nd</sup>	Hidden Surfaces removal		
14 <sup>th</sup>	53 <sup>rd</sup>	Hidden surface elimination		
	54 <sup>th</sup>	depth buffer algorithm		
	55 <sup>th</sup>	scan line coherence		
	56 <sup>th</sup>	area coherence algorithm		
15 <sup>th</sup>	57 <sup>th</sup>	priority algorithm		
	58 <sup>th</sup>	Revision		
	59 <sup>th</sup>	Test		

60<sup>th</sup>

Test