

REGULAR/BACK/SCHOLARSHIP EXAM - 2081/2082 CHAITRA/BAISHAKH

1. What is computer graphics? List out the application of computer Graphics

Computer graphics is the branch of computer science that deals with creating, manipulating, and displaying visual images using computers. It involves techniques for generating pictures, animations, and simulations on digital screens.

Applications of Computer Graphics

1. Entertainment and Animation: Used in movies, video games, and cartoons.
2. Education and Training: Interactive simulations for teaching (e.g., medical training, flight simulators).
3. Scientific Visualization: Graphs, charts, and 3D models to represent complex data.
4. Design and Engineering: CAD tools for designing machines, buildings, and electronics.
5. User Interfaces: Attractive GUIs in operating systems and applications.
6. Virtual Reality & Augmented Reality: Immersive environments for gaming and training.

computer graphics. Differentiate between CAD and CAM.

Feature	CAD (Computer-Aided Design)	CAM (Computer-Aided Manufacturing)
Full Form	Computer-Aided Design	Computer-Aided Manufacturing
Focus	Focuses on how it looks and how it is structured.	Focuses on how it is built and the process of making it.
Main Function	Creating 2D drawings and 3D models.	Controlling machines (CNC, 3D printers, Lathes).
Primary Tool	Geometrical tools, lines, and shapes.	Toolpaths, cutting speeds, and feed rates.
Output	A virtual blueprint (digital file).	A finished physical product.
Software Examples	AutoCAD, SolidWorks, Blender.	Mastercam, Fusion 360 (CAM module), PowerMill.
User	Architects, Product Designers, Engineers.	Machinists, Factory Operators, Technicians.
Dependency	It is the starting point of the process.	It usually needs a CAD file to work from.

2. Explain different types of touch panels. Differentiate between raster and vector scan display architectures.

Different Types of Touch Panels

There are several ways a screen "feels" your finger, but these three are the most common:

- **Resistive Touch Screens:** These are made of two flexible layers with a small gap between them. When you press down, the layers touch, completing an electrical circuit.

- **The Vibe:** You usually have to press a bit harder. Think of old ATM machines or checkout counters.
- **Pro:** You can use a gloved finger, a pen, or even a fingernail.
- **Con:** Not very clear and doesn't support "pinch-to-zoom."
- **Capacitive Touch Screens:** These use a layer of capacitive material that holds an electrical charge. Since the human body conducts electricity, touching the screen pulls a tiny bit of charge to your finger.
 - **The Vibe:** This is what's on your iPhone or Android. It's super responsive and smooth.
 - **Pro:** Supports multi-touch (pinching/swiping) and is very durable.
 - **Con:** Won't work with regular gloves or a plastic stylus.
- **Infrared (IR) Touch Screens:** These have a grid of invisible infrared light beams crossing the surface. When you touch the screen, you "break" the beams.
 - **The Vibe:** Often used in large kiosks or giant interactive whiteboards.
 - **Pro:** Very high clarity because there's no extra layer over the glass.
 - **Con:** Can be "tripped" by a sleeve or a bug crawling across the screen.
- Got it! Here is the 7-point comparison between **Raster Scan** and **Vector Scan** displays, presented in a clean table for your notes.
- **Difference Between Raster and Vector Scan Displays**

S.N.	Feature	Raster Scan Display	Vector Scan (Random Scan)
1	Beam Movement	The electron beam sweeps across the entire screen , line by line (top to bottom).	The electron beam moves only to the parts of the screen where a line is to be drawn.
2	Image Definition	Images are stored as a set of intensity values for discrete points called Pixels .	Images are stored as a list of mathematical Line Commands (vectors).
3	Resolution	Lower resolution; lines can look "jagged" (staircase effect) when zoomed in.	Much higher resolution; produces smooth, continuous lines and curves.
4	Color & Shading	Excellent at displaying realistic scenes, shadows, and millions of colors .	Very poor at shading; mostly used for wireframe line drawings .
5	Refresh Rate	The refresh rate is constant (e.g., 60Hz), regardless of how complex the image is.	The refresh rate decreases as the drawing becomes more complex (more lines to draw).
6	Storage	Uses a Frame Buffer to store the intensity of every single pixel on the screen.	Uses a Display List Buffer to store the specific commands for lines and points.
7	Cost	Generally cheaper and more common (used in TVs, monitors, and phones).	More expensive and specialized (used in radar, oscilloscopes, and CAD).

3. What is scaling? Write Bresenham's line drawing algorithm.

Scaling

Scaling is a basic geometric transformation in computer graphics. It means **resizing an object** either by enlarging or shrinking it.

Bresenham's Line Drawing Algorithm

Bresenham's algorithm is an efficient method to draw a straight line between two points on a raster display using only **integer arithmetic** (no floating point).

Steps (for slope between 0 and 1):

1. Input two endpoints (x_1, y_1) and (x_2, y_2) .
2. Calculate:

$$dx = x_2 - x_1, \text{quad } dy = y_2 - y_1$$

3. and initial decision parameter:

$$p = 2dy - dx$$

4. Start from $(x, y) = (x_1, y_1)$.

5. For each x from x_1 to x_2 :

- Plot point (x, y) .
- If $p < 0$:

$$p = p + 2dy$$

- $(y \text{ remains same}).$
- Else:

$$y = y + 1, \text{quad } p = p + 2dy - 2dx$$

6. Repeat until the line reaches (x_2, y_2) .

4. Given a triangle with corner coordinates $(0, 0)$, $(2, 0)$ and $(3, 2)$. Perform 2D rotation by 90° anticlockwise and find out new coordinate. Write DDA line drawing algorithm.

. 2D Rotation of the Triangle

We are rotating the triangle with corners $A(0, 0)$, $B(2, 0)$, and $C(3, 2)$ by 90° anticlockwise about the origin $(0, 0)$.

The Shortcut Rule:

For a 90° anticlockwise rotation, the rule is very simple:

Swap the x and y values, and then make the new x negative.

$$(x, y) \rightarrow (-y, x)$$

The Calculation:

1. **Point $A(0, 0)$:**

- $x' = -0 = 0$
- $y' = 0$
- **New $A' = (0, 0)$**

2. **Point $B(2, 0)$:**

- $x' = -0 = 0$
- $y' = 2$
- **New $B' = (0, 2)$**

3. **Point $C(3, 2)$:**

- $x' = -2$
- $y' = 3$
- **New $C' = (-2, 3)$**

Result: After the 90° rotation, your triangle moves from the bottom-right of the graph toward the top-left. Your new coordinates are $(0, 0)$, $(0, 2)$, and $(-2, 3)$.

DDA line drawing algorithm (exam-style steps)

1. **Compute deltas:**

$$dx = x_2 - x_1, \quad dy = y_2 - y_1$$

2. **Choose steps:**

$$\text{steps} = \max(|dx|, |dy|)$$

3. **Increments per step:**

$$x_{\text{inc}} = \frac{dx}{\text{steps}}, \quad y_{\text{inc}} = \frac{dy}{\text{steps}}$$

4. **Initialize:**

$$x = x_1, \quad y = y_1$$

5. **Iterate and plot:**

Repeat for steps times:

- Plot $(\text{round}(x), \text{round}(y))$.
- Update:

$$x = x + x_{\text{inc}}, \quad y = y + y_{\text{inc}}$$

6. **Stop** when the loop completes—line reaches (x_2, y_2) .

5. Write the matrix notation for shearing. Explain different steps involved in 3D viewing pipeline.

Matrix Notation for Shearing

Shearing is a transformation that distorts the shape of an object by shifting its points in a fixed direction, depending on their coordinate values.

For 2D Shearing, the matrix form is:

- S_x = shear factor in x-direction
- S_y = shear factor in y-direction

👉 If, the object is slanted horizontally.

👉 If, the object is slanted vertically.

Steps in 3D Viewing Pipeline

The 3D viewing pipeline is the sequence of steps used to convert a 3D scene into a 2D image on the screen.

1. **Modeling Transformation**

- Converts object coordinates into world coordinates.
- Example: placing a car model into a city scene.

2. **Viewing Transformation**

- Defines the position and orientation of the camera (eye coordinates).
- World coordinates $\rightarrow$ camera coordinates.

3. **Projection Transformation**

- Converts 3D camera coordinates into 2D projection coordinates.

• **Types:**

- Parallel Projection (orthographic) $\rightarrow$ preserves shape.
- Perspective Projection $\rightarrow$ gives realistic depth effect.

4. **Clipping**

- Removes objects or parts of objects outside the viewing volume (not visible to the camera).

5. **Viewport Transformation**

- Maps the 2D projection coordinates onto the actual screen (device coordinates).
- Ensures the final image fits the display window.

6. Define perspective projection. Differentiate between object space and image space method in hidden surface removal

perspective Projection

Perspective projection is a technique in computer graphics that converts 3D objects into a 2D view while giving a sense of depth. It mimics how the human eye sees the world: objects that are farther away appear smaller, and parallel lines seem to converge at a point (vanishing point)

Differentiate between object space and image space method in hidden surface removal

S.N.	Feature	Object Space Method	Image Space Method
1	Basic Philosophy	It asks: "Which object is in front of the other in the 3D world ?"	It asks: "Which object is visible at this specific pixel ?"
2	Coordinate System	Works in Physical/World Coordinates (\$x, y, z\$).	Works in Screen/Device Coordinates (\$x, y\$).
3	When it's done	Performed early in the pipeline, before projection.	Performed late in the pipeline, during scan conversion.
4	Precision	Infinite precision (mathematically exact).	Limited by the display resolution (pixel size).
5	Complexity Factor	Depends on the number of objects (\$n\$) in the scene.	Depends on the number of pixels (\$p\$) on the screen.
6	Scalability	Gets very slow if you add thousands of objects.	Speed stays mostly the same even if objects increase.
7	Memory Usage	Low; only needs to store the object geometry.	High; needs a Z-Buffer to store depth for every pixel.
8	Calculations	Involves heavy geometric comparisons (lines/planes).	Involves simple depth (\$z\$) comparisons at each pixel.
9	Zooming Effect	If you zoom in, the lines remain perfectly sharp.	If you zoom in, you might see "aliasing" or jagged edges.
10	Famous Examples	Back-Face Culling , Area Subdivision (Warnock).	Z-Buffer (Depth Buffer) , Ray Casting, Scan-line

7. What is parallel projection? Explain constant shading and phong shading model.

parallel projection is a method in computer graphics where 3D objects are projected onto a 2D plane using parallel lines. Unlike perspective projection, the size of objects does not change with distance — so there is no vanishing point.

Constant Shading (Flat Shading)

Constant shading, also called flat shading, is the simplest shading technique in computer graphics.

Process

- Each polygon surface is assigned a single intensity value.
- The color is constant across the entire polygon.
- The intensity is usually calculated using the polygon's normal vector and a lighting model.

Advantages

- Very simple and fast to compute.
- Good for wireframe or low-detail models.

Disadvantages

- Surfaces look unrealistic.

- Edges between polygons are clearly visible (faceted look).
- Cannot represent smooth curves or highlights.

Phong Shading Model

Phong shading is a more advanced shading technique that produces smooth and realistic surfaces.

Process

1. Normal Interpolation:
 - Normal vectors are calculated at each vertex of the polygon.
 - These normals are interpolated across the surface.
2. Illumination Calculation:
 - At each pixel, the interpolated normal is used in the Phong illumination model.
 - The model considers ambient, diffuse, and specular reflection.
3. Specular Highlights:
 - Phong shading can show shiny spots on surfaces, making them look realistic.

Advantages

- Produces smooth shading across surfaces.
- Captures specular highlights (shiny reflections).
- Much more realistic than flat or Gouraud shading.

Disadvantages

- Computationally expensive compared to flat shading.
- Requires more processing power and memory

8. What are different graphics file formats? Explain different features of graphics packages.

Different Graphics File Formats

Graphics formats are generally divided into two main categories: **Raster** (pixel-based) and **Vector** (math-based).

A. Raster Formats (Pixels)

- **JPEG/JPG:** Best for photos. It uses "lossy" compression, meaning it shrinks the file size by throwing away some tiny details you won't notice.
- **PNG:** Great for web graphics. It supports **transparency** and "lossless" compression (no quality loss).
- **GIF:** Limited to 256 colors. It's famous for simple animations and is also lossless.
- **TIFF:** Used in professional printing and photography because it keeps every bit of data (very high quality, but huge files).

B. Vector Formats (Paths)

- **SVG:** The standard for web icons and logos. You can zoom in forever, and it never gets blurry.
- **EPS/AI:** Used by professionals (like in Adobe Illustrator) for high-end design and printing.
- **Explain different features of graphics packages.**

features of Graphics Packages

A "Graphics Package" is just a fancy name for software like Photoshop, CorelDraw, or AutoCAD. To be useful, these programs usually come with these core features:

1. Primitive Creation

The software must let you draw basic shapes (primitives) like lines, circles, rectangles, and polygons. You shouldn't have to code them; you just click and drag.

2. Transformation Tools

This allows you to manipulate what you've drawn. Key transformations include:

- **Translation:** Moving the object.
- **Rotation:** Turning the object.
- **Scaling:** Changing the size.

3. Attribute Control This is where you change the "look" of the objects. You can change line thickness, color, fill patterns, and transparency.

9. What do you mean by virtual reality? List out its application. Differentiate between semi-immersive and fully immersive virtual reality’.

Virtual Reality is a computer-generated simulation that allows users to experience and interact with a 3D environment as if it were real. By using devices like head-mounted displays (HMDs), gloves, or motion sensors, VR creates a sense of presence, making the user feel immersed inside the virtual world.

Applications of Virtual Reality

1. Gaming and Entertainment – Immersive video games, 3D movies, theme park rides.
2. Education and Training – Medical surgery practice, flight simulators, military drills.
3. Architecture and Design – Virtual walkthroughs of buildings before construction.
4. Healthcare – Therapy for phobias, rehabilitation exercises.
5. Tourism – Virtual tours of historical sites and museums.
6. Business and Collaboration – Virtual meetings, product demos.

Differentiate between semi-immersive and fully immersive virtual reality’.

S.N.	Feature	Semi-Immersive VR	Fully Immersive VR
1	Definition	You are focused on a virtual world but stay connected to the real one.	You are completely cut off from the physical world.
2	Hardware	Large screens, high-end projectors, or monitors.	Head-Mounted Displays (HMDs) and noise-canceling headphones.
3	View	Provides a limited field of view (usually 2D or 3D on a screen).	Provides a 360-degree view of the virtual environment.
4	Awareness	You can still see your own body and the room around you.	You see only your "avatar" or virtual hands; the room is invisible.
5	Interaction	Uses physical tools like a steering wheel, joystick, or mouse.	Uses motion sensors, gloves, and body tracking.
6	Immersion	Moderate; you feel like you are "watching" a scene.	High; you feel like you are "inside" the scene.
7	Example	A high-end flight simulator used by pilots.	Playing a VR game (e.g., <i>Half-Life: Alyx</i>) with an Oculus/Meta Quest.

10. Write short notes on.

a. CRT Monitors

- Definition: CRT (Cathode Ray Tube) monitors are the traditional display devices that use electron beams to excite phosphors on the screen, producing images.
- **Working Principle:**
- Electron gun emits beams.
- Magnetic/electric fields deflect beams horizontally and vertically.

- Phosphor coating glows when struck by electrons.
- Features:
- Bulky and heavy.
- Good color reproduction and viewing angles.
- High power consumption.
- Use: Widely used before LCD/LED monitors became popular.

b. Polygon Tables

- Definition: Polygon tables are data structures used in computer graphics to represent objects made of polygons.
- Contents:
- Vertex list (coordinates of corners).
- Edge list (connections between vertices).
- Surface properties (color, shading, texture).
- Purpose:
- Helps in rendering, hidden surface removal, and shading.
- Efficient way to store and process 3D objects.

c. Browser Safe Colors

- Definition: Browser safe colors are a palette of 216 colors that display consistently across different web browsers and computer systems.
- Reason: Early monitors supported only 256 colors; overlapping palettes gave 216 common colors.
- Features:
- Represented using hex values (#RRGGBB).
- Each component (R, G, B) takes values: 00, 33, 66, 99, CC, FF.
- Use: Ensures web pages look the same on all systems, especially older browsers.
- Modern Note: With today's displays, full 24-bit color is supported, so browser safe colors are less critical

REGULAR EXAM-2081 JESTHA/ASHADH

1. Define computer graphics and its application in different fields. Differentiate between raster and random scan display.

Computer Graphics is the branch of computer science that deals with creating, manipulating, and displaying visual content using computers. It involves generating images, diagrams, and animations from data and mathematical models.

Applications of Computer Graphics.

1. **Education & Training**
 - Interactive learning tools, simulations (e.g., medical surgery practice, flight simulators).
2. **Entertainment**
 - Video games, movies, animations, special effects.
3. **Engineering & Design (CAD/CAM)**
 - Designing cars, buildings, machines using CAD software.
4. **Medical Field**
 - Visualizing CT scans, MRI images, 3D anatomy models.
5. **Scientific Visualization**
 - Graphs, charts, weather maps, molecular structures.
6. **Business & Advertising**
 - Presentations, infographics, product demos, digital marketing.
7. **Virtual Reality & Augmented Reality.**
 - Immersive environments for gaming, tourism, and training.

Differentiate between raster and random scan display.

S.N.	Feature	Raster Scan Display	Random (Vector) Scan Display
1	Beam Movement	Sweeps the entire screen, row by row, from top to bottom.	The beam moves only to the parts where the image is to be drawn.
2	Image Storage	Stored as a set of pixels in a Frame Buffer .	Stored as a list of line-drawing commands in a Display List .
3	Resolution	Lower; produces "jagged" lines (staircase effect) on diagonals.	Much higher; produces smooth, continuous lines .
4	Color & Shading	Excellent for realistic scenes and millions of colors.	Poor; mostly limited to simple wireframe line drawings.

S.N.	Feature	Raster Scan Display	Random (Vector) Scan Display
5	Refresh Rate	Constant (e.g., 60Hz), no matter how simple the image is.	Depends on the number of lines; the more lines, the slower it refreshes.
6	Cost	Generally cheaper and more common (standard TVs/PCs).	Highly expensive and specialized.
7	Applications	Photos, movies, and modern desktop interfaces.	Radar, oscilloscopes, and early high-end CAD stations.

2. Using midpoint circle algorithm plot a circle whose radius is equal to 10, centered at origin.

For a circle with radius $r = 10$ and center $(0,0)$, the **Midpoint Circle Algorithm** calculates the pixels for one octant (from $x=0$ to $x=y$) and then uses symmetry to plot the rest.

1. Initial Values

- **Radius (r):** 10
- **Starting Point:** $(0, r) = (0, 10)$
- **Initial Decision Parameter (P_0):**

$$P_0 = 1 - r = 1 - 10 = -9$$

2. Iteration Table (First Octant)

Step (k)	P_k	Next Pixel (x_{k+1}, y_{k+1})	Condition	P_{k+1} Calculation
0	-9	(0, 10)	$P_k < 0$	$P_{k+1} = -9 + 2(0) + 3 = -6$
1	-6	(1, 10)	$P_k < 0$	$P_{k+1} = -6 + 2(1) + 3 = -1$
2	-1	(2, 10)	$P_k < 0$	$P_{k+1} = -1 + 2(2) + 3 = 6$
3	6	(3, 10)	$P_k \geq 0$	$P_{k+1} = 6 + 2(3 - 10) + 5 = -3$
4	-3	(4, 9)	$P_k < 0$	$P_{k+1} = -3 + 2(4) + 3 = 8$
5	8	(5, 9)	$P_k \geq 0$	$P_{k+1} = 8 + 2(5 - 9) + 5 = 5$
6	5	(6, 8)	$P_k \geq 0$	$P_{k+1} = 5 + 2(6 - 8) + 5 = 6$
7	6	(7, 7)	Stop	$x = y$, octant complete.

3. Symmetry (Plotting the full circle)

For every point (x, y) calculated above, we plot 8 points on the grid:

$$(x, y), (y, x), (-x, y), (-y, x), (-x, -y), (-y, -x), (x, -y), (y, -x)$$

Example for Step 4 (4, 9):

Points: $(4, 9), (9, 4), (-4, 9), (-9, 4), (-4, -9), (-9, -4), (4, -9), (9, -4)$

3. Derive the DDA line drawing algorithm for $m < 1$ and $m > 1$.

DA Line Drawing Algorithm

The **Digital Differential Analyzer (DDA)** algorithm is a **scan-conversion line-drawing algorithm** based on calculating either Δx or Δy using the slope of the line. It works by **incrementing either x or y by 1** and calculating the other coordinate.

Basic Idea:

For a line from (x_1, y_1) to (x_2, y_2) :

- If slope $|m| < 1 \rightarrow$ **Increment x by 1**, calculate y
- If slope $|m| > 1 \rightarrow$ **Increment y by 1**, calculate x

1. Derivation for Slope $|m| < 1$

Given:

- Line from (x_1, y_1) to (x_2, y_2)
- Slope $m = (y_2 - y_1) / (x_2 - x_1)$
- Condition: $|m| < 1$ (i.e., $|\Delta y| < |\Delta x|$)

Derivation:

The line equation is:

$$y = mx + c \Rightarrow y = mx + c$$

Where:

$$m = \frac{\Delta y}{\Delta x} = \frac{y_2 - y_1}{x_2 - x_1} \quad m = \frac{\Delta y}{\Delta x} = \frac{y_2 - y_1}{x_2 - x_1}$$

For each step:

1. Increment **x by 1** (since $\Delta x > \Delta y$ in magnitude)
2. Calculate new y:

$$y_{i+1} = m \cdot x_{i+1} + c \Rightarrow y_{i+1} = m \cdot x_{i+1} + c$$

$$\text{But } x_{i+1} = x_i + 1 \Rightarrow x_{i+1} = x_i + 1$$

So:

$$y_{i+1} = m(x_i + 1) + c = (mx_i + c) + m = y_i + m \Rightarrow y_{i+1} = y_i + m$$

Thus:

$$x_{i+1} = x_i + 1 \quad x_{i+1} = x_i + 1 \quad y_{i+1} = y_i + m \quad y_{i+1} = y_i + m$$

$$\text{Where } m = \frac{y_2 - y_1}{x_2 - x_1} \quad m = \frac{y_2 - y_1}{x_2 - x_1}$$

Algorithm Steps for $|m| < 1$:

1. Calculate Δx , Δy , m
2. Start from (x_1, y_1)

3. Plot point (round(x), round(y))
4. Update: $x = x + 1$, $y = y + m$
5. Repeat until x reaches x_2

2. Derivation for Slope $|m| > 1$

Given:

- Same line equation
- Condition: $|m| > 1$ (i.e., $|\Delta y| > |\Delta x|$)

Derivation:

Rewrite line equation in terms of x:

$$x = y - cm \quad x = my - c$$

Or from slope formula:

$$m = \frac{\Delta y}{\Delta x} \Rightarrow \Delta x = \frac{\Delta y}{m} \quad m = \frac{\Delta x}{\Delta y} \Rightarrow \Delta x = m \Delta y$$

Since $|m| > 1$, it's better to **increment y by 1** and calculate x.

For each step:

1. Increment **y by 1**
2. Calculate new x:

$$x_{i+1} = y_{i+1} - cm \quad x_{i+1} = my_{i+1} - c$$

$$\text{But } y_{i+1} = y_i + 1 \quad y_{i+1} = y_i + 1$$

So:

$$x_{i+1} = (y_i + 1) - cm = y_i - cm + 1m = x_i + 1m \quad x_{i+1} = m(y_i + 1) - c = my_i - c + m1 = x_i + m1$$

Thus:

$$y_{i+1} = y_i + 1 \quad y_{i+1} = y_i + 1 \quad x_{i+1} = x_i + 1m \quad x_{i+1} = x_i + m1$$

Algorithm Steps for $|m| > 1$:

1. Calculate Δx , Δy , m
2. Start from (x1, y1)
3. Plot point (round(x), round(y))
4. Update: $y = y + 1$, $x = x + 1/m$
5. Repeat until y reaches y_2 .

4. What is perspective projection? Explain 3D reflection and rotation with example.

Perspective Projection.

Perspective projection is a technique in computer graphics that converts 3D objects into a 2D view while giving a sense of depth. It mimics how the human eye sees the world: objects farther away appear smaller, and parallel lines seem to converge at a vanishing point.

3D Reflection (In-Depth)

Reflection sirf flip nahi hai, ye actually ek **180-degree rotation** jaisa hota hai jo 4th dimension ke through visualize kiya ja sakta hai.

- **Plane of Symmetry:** 3D mein reflection hamesha ek plane ke relative hota hai.
- **Sign Change Logic:**
 - **XY Plane Reflection:** Mirror $Z=0$ par hai. Isliye sirf Z coordinate apni direction badalta hai ($z \rightarrow -z$).
 - **YZ Plane Reflection:** Mirror $X=0$ par hai. Isme x coordinate badal jata hai ($x \rightarrow -x$).
 - **XZ Plane Reflection:** Mirror $Y=0$ par hai. Isme y coordinate badal jata hai ($y \rightarrow -y$).
- **Example for Exam:** Agar ek cube $(1,1,1)$ par hai, aur hum use **YZ plane** par reflect karein, toh uski position $(-1,1,1)$ ho jayegi. Iska matlab object left side se right side (ya vice versa) shift ho gaya.

3. 3D Rotation (Detailed Calculation)

3D mein rotation 2D se mushkil isliye hai kyunki humein **Axis** choose karni padti hai.

- **Right-Hand Rule:** Agar aap apne anguthe (thumb) ko rotation axis ki taraf rakhein, toh apki ungliyan (fingers) jis taraf ghumengi, wo **Positive Rotation** (θ) hoga.
- **Coordinate Stability:** Jis axis ke around rotation ho raha hai, uski value kabhi change nahi hoti.
 - **Rotation about X-axis:** x constant rahega. y aur z change honge.
 - **Rotation about Y-axis:** y constant rahega. x aur z change honge.
- **Deep Formula (Y-axis Rotation):** Isme thoda dhyan dena padta hai kyunki iska sign pattern X aur Z se alag hota hai:
 - $x' = x \cdot \cos(\theta) + z \cdot \sin(\theta)$
 - $z' = -x \cdot \sin(\theta) + z \cdot \cos(\theta)$
 - $y' = y$.

Example:

Point $(1,0,0)$ ko 90° Z-axis ke around rotate karne par $\rightarrow (0,1,0)$.

5. What are different hidden surface removal techniques? Explain.

Hidden Surface Removal (HSR)

Hidden surface removal is the process of identifying and removing surfaces of 3D objects that are **not visible** from the viewer's position, so that only the **visible parts** are displayed on the screen.

Different Hidden Surface Removal Techniques

1. Back-Face Removal Method

Explanation

- A surface is hidden if it faces **away from the viewer**
- Such surfaces are not drawn

Working

- Calculate the **angle between surface normal and viewing direction**
- If angle $> 90^\circ$, the surface is removed

Advantages

- Simple and fast
- Reduces processing time

Limitation

- Works only for **closed objects**

2. Z-Buffer (Depth Buffer) Algorithm**Explanation**

- Uses a **depth (Z) value** for each pixel
- Stores the depth of the **closest surface**

Working

1. Initialize Z-buffer with large depth values
2. Compare depth of each pixel
3. Display the pixel with **minimum Z value**

Advantages

- Simple to implement
- Works with complex scenes

Limitation

- Requires extra memory

3. Painter's Algorithm**Explanation**

- Surfaces are drawn from **back to front**
- Closer surfaces overwrite distant ones

Working

- Sort surfaces by depth
- Draw farthest surface first

Advantages

- Easy concept
- Good for simple scenes

Limitation

- Fails when surfaces intersect

4. Scan-Line Algorithm**Explanation**

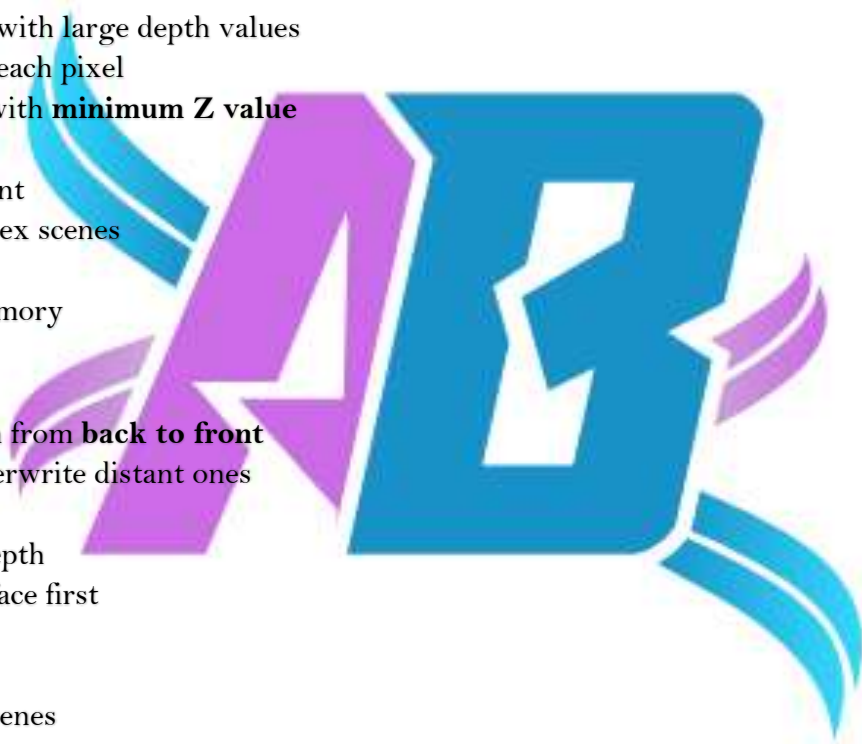
- Processes one **horizontal scan line** at a time
- Determines visible surface on each scan line

Working

- Find intersection of scan line with polygons
- Compare depth values
- Display nearest surface

Advantages

- Faster than Z-buffer
- Less memory usage



5. Explain phong shading model. Describe different types of virtual reality.

Phong Shading Model.

Phong shading is a technique in computer graphics used to produce smooth and realistic surfaces by interpolating surface normals and applying the Phong illumination model at each pixel.

Steps in Phong Shading

1. **Normal Calculation:**
 - Normal vectors are computed at each vertex of the polygon.
2. **Normal Interpolation:**
 - These normals are interpolated across the surface of the polygon.
3. **Illumination Calculation:**
 - At each pixel, the interpolated normal is used in the Phong illumination equation:
 - where:
 - = Ambient reflection (background light)
 - = Diffuse reflection (matte surface light)
 - = Specular reflection (shiny highlights)

Features

- Produces smooth shading across surfaces.
- Captures specular highlights (shiny spots).
- More realistic than flat shading or Gouraud shading.
- Computationally heavier, but widely used in modern rendering.

Types of Virtual Reality (VR)

Virtual Reality is classified based on the degree to which the user feels "immersed" or present in the virtual environment.

1. Fully Immersive VR

This provides the highest level of realism where the user is completely cut off from the physical world.

- **Experience:** The user gets a 360° field of view. Every movement of the head or body is tracked and reflected in the virtual world, making it feel like you are actually "inside" the simulation.
- **Hardware:** Requires a **Head-Mounted Display (HMD)** (e.g., Oculus Rift, HTC Vive), specialized motion-tracking sensors, and noise-canceling headphones to block out real-world stimuli.
- **Example:** High-end VR gaming, virtual surgery for medical students, or deep-sea exploration simulations.

2. Semi-Immersive VR

This type provides a partial sense of being in another environment while the user remains aware of their real-world surroundings.

- **Experience:** It typically uses large screens or high-resolution projections rather than a headset. You can interact with the virtual world, but you can still see your own body and the room you are in.
- **Hardware:** Large-scale projection systems, high-performance graphics workstations, and sometimes physical mock-ups (like a car or cockpit).
- **Example: Flight Simulators** used for pilot training. The cockpit and controls are real, but the view outside the "windows" is a high-quality 3D projection.

3. Non-Immersive VR

This is the most common form of VR, where the virtual environment is viewed through a standard screen.

- **Experience:** The user interacts with a 3D environment but remains fully grounded in the physical world. There is no sensory immersion; you are simply an observer controlling a character or viewpoint.
- **Hardware:** Standard desktop computers, laptops, or consoles using a monitor, mouse, keyboard, or controller.
- **Example:** Playing 3D video games (like GTA or Counter-Strike) or using Google Street View to explore a city.

6. Define the terms resolution and anti-aliasing. What are the different graphics packages and mention their features?.

Definition: Resolution refers to the number of pixels displayed on the screen horizontally and vertically. It determines the clarity and detail of an image. Higher resolution → sharper and more detailed image.

Anti-Aliasing

- **Definition:** Anti-aliasing is a technique used in computer graphics to smooth jagged edges (aliasing) that appear when displaying diagonal or curved lines on pixel grids. It works by blending edge pixels with background colors to create a smoother transition.

Graphics Packages and Their Features

Graphics packages are software tools used to create, edit, and manipulate images.

1. Adobe Photoshop (Raster Graphics)

- Pixel-based editing.
- Features: Layers, filters, photo retouching, color correction, text effects.
- Best for photo editing and digital painting.

2. CorelDRAW (Vector Graphics)

- Shape-based editing.
- Features: Vector drawing tools, typography, logo design, scalable graphics.
- Best for logos, posters, and print media.

3. AutoCAD (CAD Graphics)

- Engineering and architectural design.
- Features: 2D drafting, 3D modeling, precision tools, technical drawings.
- Best for mechanical, civil, and architectural projects.

4. Blender (3D Graphics)

- Open-source 3D modeling and animation.
- Features: Modeling, rendering, animation, simulation, video editing.
- Best for 3D animation, movies, and game design.

5. MS Paint / Paint.NET (Basic Graphics)

- Simple raster graphics editing.
- Features: Drawing tools, basic shapes, color fills.
- Best for beginners and simple tasks.

8. Describe different types of input and output devices used in computer graphics.

1. Input Devices

These devices allow users to communicate with the computer to draw, select, or modify graphical data.

- **Keyboard:** The most basic input device used to enter commands, coordinates, or text labels into a graphics program.
- **Mouse:** A point-and-click device used for selecting icons, drawing lines, and navigating 2D interfaces.
- **Joystick:** Consists of a stick that pivots on a base. It is primarily used for 3D navigation and controlling movement in flight simulators or games.
- **Light Pen:** A stylus-like device that is sensitive to light. A user can point directly at the screen to select objects or draw on the monitor.
- **Graphics Tablet (Digitizer):** A flat pad accompanied by a stylus. It is highly popular among digital artists as it can detect pressure sensitivity, allowing for realistic brush strokes.
- **Scanner:** Used to convert physical images, photos, or documents into digital raster formats that the computer can process.
- **Trackball:** Similar to a mouse but with the ball on top. The user rotates the ball with their fingers, which is useful when desk space is limited.
- **Data Glove:** Used in Virtual Reality (VR), this glove captures hand movements and finger positions to interact with 3D virtual objects.

2. Output Devices

These devices translate digital data into a visual form that a human can see.

- **Monitors (CRT/LCD/LED):** The primary output device. Modern LED/LCD monitors use a matrix of pixels to display high-resolution graphics.
- **Printers:** Used to create "hard copies" of graphics on paper.
 - **Inkjet/Laser:** Common for 2D images.
 - **3D Printers:** Used to create physical 3D objects from digital models.
- **Plotters:** Specialized devices used for printing high-quality vector graphics, such as architectural blueprints or engineering drawings. Unlike printers, they use pens to draw continuous lines.
- **Data Projectors:** Used to project computer images onto a large screen for presentations or semi-immersive VR setups.
- **Head-Mounted Display (HMD):** Used in Fully Immersive VR to provide a separate image to each eye, creating a 3D stereoscopic effect.

9. Write short notes on.

a. Two Dimensional (2D) Transformation

2D Transformation is the process of modifying or repositioning a 2D object on a coordinate plane. These are essential for animation and computer-aided design (CAD).

There are three basic types:

1. **Translation:** Moving an object from one position (x, y) to another (x', y') by adding a displacement value (t_x, t_y) .
 - $x' = x + t_x$
 - $y' = y + t_y$
2. **Scaling:** Changing the size of an object. This is done by multiplying the coordinates by scaling factors (s_x, s_y) .
 - $x' = x \cdot s_x$
 - $y' = y \cdot s_y$
3. **Rotation:** Turning an object around a fixed point (usually the origin) by an angle θ .
 - $x' = x \cos \theta - y \sin \theta$
 - $y' = x \sin \theta + y \cos \theta$

b. Polygon Table

A **Polygon Table** is a data structure used in computer graphics to store information about 3D objects made of polygons (like triangles or quadrilaterals). Instead of storing every point multiple times, it uses a organized system of three tables:

1. **Vertex Table:** Stores the actual coordinates (x, y, z) for every unique corner (vertex) of the object.
2. **Edge Table:** Lists which vertices connect to form an edge. For example, Edge E_1 connects Vertex V_1 and V_2 .
3. **Polygon-Surface Table:** Lists which edges connect to form a face or surface.

Why use it? It is highly efficient. If you move one vertex, all connected edges and surfaces update automatically without the computer having to re-calculate every single point.

c. CRT Monitor (Cathode Ray Tube)

The **CRT Monitor** was the standard display device for computers for many years before LCDs. It uses a vacuum tube to project images.

- **How it Works:** At the back, an **electron gun** fires a beam of high-speed electrons. A magnetic field (deflection yoke) directs this beam across the screen.
- **Phosphor Screen:** The inside of the front glass is coated with tiny dots of **phosphor**. When the electron beam hits these dots, they glow to create pixels.
- **Refresh Rate:** Because the phosphor glow fades quickly, the electron gun must "redraw" the image many times per second (usually 60–80 times) so the human eye doesn't see a flicker.
- **Color CRT:** For color, there are three electron guns (Red, Green, Blue) and a **shadow mask** to ensure each beam hits the correct color dot.

REGULAR/BACK EXAM-2080, BHADRA.

1. What is computer graphics? Explain application of computer graphics.

Computer Graphics as the bridge between raw data and human vision. In simple terms, it is the technology used to create, store, and manipulate images and geometric structures using a computer. It allows us to represent complex information visually, making it easier to understand, analyze, and interact with.

Applications of Computer Graphics

Computer graphics is used in almost every industry today. Here are the most prominent fields:

1. Computer-Aided Design (CAD)

Architects and engineers use graphics to create precise 3D models of buildings, cars, and machine parts. This allows them to test structures virtually before they are built in the real world.

2. Entertainment (Movies and Gaming)

This is the most visible application. From the CGI (Computer Generated Imagery) in blockbuster movies to the immersive 3D worlds of modern video games, computer graphics is used to create realistic characters, environments, and special effects.

3. Medical Imaging

Doctors use graphics to turn data from MRI, CT scans, and X-rays into 3D visualizations. This helps in planning surgeries and diagnosing internal health issues without invasive procedures.

4. Education and Training

Simulators are used to train people in high-stakes jobs. For example, flight simulators use real-time graphics to train pilots in a safe environment, and medical simulators allow students to practice surgeries.

5. Business Graphics (Data Visualization)

Complex data is turned into easy-to-read charts, graphs, and 3D maps. This is essential for financial reporting, market analysis, and presentations where "seeing" a trend is better than reading a table of numbers.

2. Define graphics input and output hardware. Differentiate between Raster and Vector display.

Graphics Input Hardware .

Input hardware wo devices hote hain jo user ke ideas, commands, aur images ko computer ke andar feed karte hain. Ye devices allow karte hain ki hum graphics ko create, manipulate aur interact kar saken.

Graphics Output Hardware.

Output hardware wo devices hote hain jo computer ke generated graphics ko user ke saamne display karte hain. Ye devices visual representation provide karte hain

Comparison: Raster Scan vs. Vector (Random Scan) Display

S.No	Feature	Raster Scan Display	Vector (Random Scan) Display
1	Basic Definition	It represents images as a grid of tiny dots called pixels.	It represents images as mathematical line segments.
2	Beam Movement	The electron beam moves across the whole screen, row by row (top to bottom).	The electron beam moves only to parts of the screen where a line is drawn.
3	Resolution	Lower resolution compared to vector; images can look "jagged" or pixelated.	Very high resolution; lines are perfectly smooth and sharp.
4	Memory Usage	Uses a Frame Buffer to store the intensity/color of every single pixel.	Uses a Display List to store commands like "Draw Line from A to B".
5	Cost	Generally cheaper and mass-produced (TVs, Monitors).	More expensive and used for specialized scientific/engineering work.
6	Realism	Excellent for displaying realistic photos with complex shading and colors.	Poor at realism; limited to wireframe models and simple line art.
7	Refresh Rate	The refresh rate is constant (usually 60Hz or higher) regardless of image complexity.	Refresh rate decreases as the number of lines increases, leading to flicker.
8	Modification	To change an image, you must update the entire grid of pixels.	To change an image, you only need to update the specific line command in the display list

3. What is Two-dimensional transformation? Explain line drawing algorithms.

Two-Dimensional Transformation

A transformation in computer graphics means changing the position, size, orientation, or shape of an object. In 2D graphics, transformations are applied to points, lines, and shapes in the plane.

Line Drawing Algorithms

Line drawing algorithms are used to plot straight lines on raster displays (pixel grids). Since pixels are discrete, algorithms approximate the line using nearest pixels.

1. Digital Differential Analyzer (DDA) Algorithm

- Works by incrementing one coordinate (x or y) in unit steps and calculating the other using slope.
- Steps:
 1. Compute , .
 2. Steps = $\max(dx, dy)$.
 3. Increment: , .
 4. Plot points by repeatedly adding increments.
- Advantage: Simple, easy to implement.
- Disadvantage: Uses floating point $\rightarrow$ slower.

👉 Example: Line from (0,0) to (8,4).

- $dx=8, dy=4 \rightarrow \text{steps}=8$.
- $x_inc=1, y_inc=0.5$.
- Points: (0,0), (1,0.5), (2,1), ... (8,4).

2. Bresenham's Line Algorithm

- Uses integer arithmetic (no floating point).
- Decision parameter decides whether next pixel is directly east or northeast.
- Steps:
 1. Start at .
 2. Compute decision variable .
 3. If : next pixel = (x+1, y).
- If : next pixel = (x+1, y+1), update .
- Advantage: Fast, efficient, widely used.

- Disadvantage: More complex logic than DDA.

👉 Example: Line from (0,0) to (8,4).

- $dx=8, dy=4 \rightarrow p=2dy - dx = 0.$
- Points chosen step by step using decision variable.

4. What is lightening model? What is parallel and perspective projection.

Lighting Model

A lighting model (illumination model) in computer graphics defines how light interacts with surfaces. It calculates brightness and color at each point using ambient, diffuse, and specular reflection. Ambient gives background light, diffuse gives matte effect, and specular gives shiny highlights. The Phong model is the most common, combining all three for realistic shading.

It is used to produce smooth shading and realistic visual effects in 3D scenes.

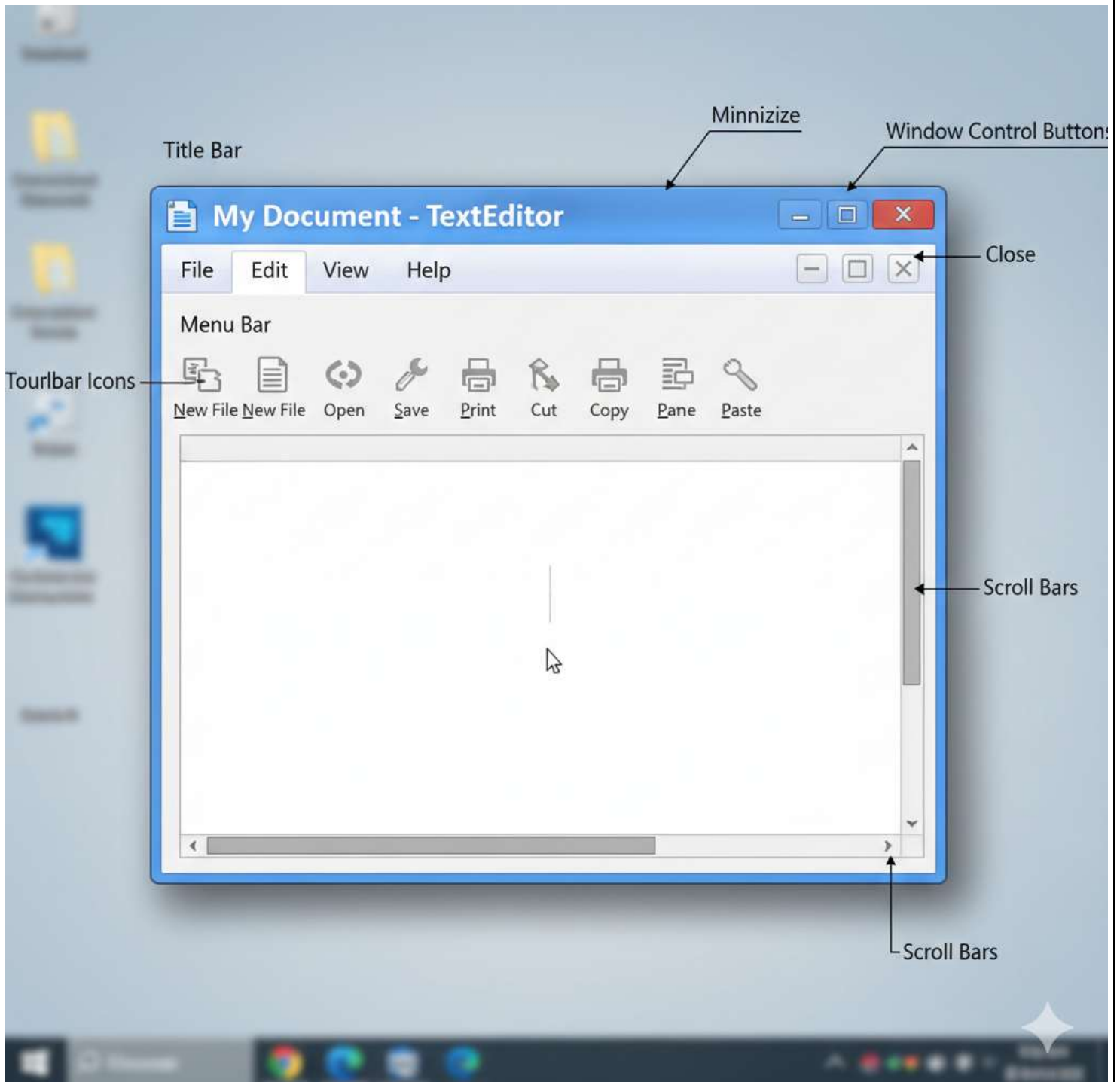
Parallel Projection

Parallel projection is a method where projectors are parallel to each other. Objects are projected onto the view plane without perspective distortion. True size and shape of objects are preserved regardless of distance. It is mainly used in engineering drawings and CAD systems. Types include orthographic and oblique projection.

Perspective Projection

Perspective projection is a method where projectors converge at a single point. Objects farther from the viewer appear smaller, creating a depth effect. Parallel lines seem to meet at a vanishing point, mimicking human eye vision. It produces realistic images with proper depth perception. Commonly used in games, movies, and virtual simulations..

5. Draw and explain the function of windows icons, menu and graphical items found on window.



1. Icons

Icons are small pictures or symbols that represent a program, a file, or a folder.

- **Function:** They act as visual shortcuts. Instead of having to remember and type a long command to open a program, you can just find its icon and double-click

it. For example, a blue 'W' icon on your desktop likely represents Microsoft Word. Icons make the computer's content easy to identify and access quickly.

2. Menu Bars

A menu bar is a horizontal bar, usually located just below the window's title bar, that contains a list of text labels (like "File," "Edit," and "View").

- **Function:** A menu bar organizes commands and functions into logical categories. When you click on a menu label, a "drop-down menu" appears with a list of related options. For example, under the "File" menu, you would find options for "New," "Open," "Save," and "Print." This system helps you find and use all the features of a program without having to memorize everything.

3. Graphical Items (Widgets)

Beyond icons and menus, windows are filled with a variety of other graphical items that help you interact with the content. These are often called "widgets" or "controls."

- **Window Control Buttons:** These are the small buttons in the top-right corner of the window.
 - **Minimize (-):** Hides the window from view, shrinking it to an icon on the taskbar. The program is still running.
 - **Maximize/Restore (square):** Maximizing makes the window fill the entire screen. When the window is already maximized, the button changes to a "restore" icon (often two overlapping squares) that returns the window to its previous size.
 - **Close (X):** Shuts down the program and closes the window.
- **Scroll Bars:** These are vertical or horizontal bars with a slider that appear on the side or bottom of a window.
 - **Function:** They allow you to "scroll" through content that is too large to fit in the window at once. You can click on the arrows at either end of the bar or drag the slider to move the content up, down, left, or right.
- **Toolbars:** These are rows of icons, often located below the menu bar.

Function: They provide quick, one-click access to the most common commands. For example, a toolbar in a word processor might have icons for "Save," "Print," "Bold," and "Italic." They are essentially shortcuts for the commands found in the menu bar

6. Explain principles of web graphic design.

Principles of Web Graphic Design.

1. Balance

- Ensures visual stability by distributing elements evenly.
- Can be symmetrical (formal, professional) or asymmetrical (dynamic, modern).
- Example: Equal weight of text and images on opposite sides of a page.

2. Contrast

- Highlights differences between elements (color, size, shape, texture).
- Makes important content stand out.
- Example: Dark text on a light background for readability.

3. Emphasis

- Directs user attention to the most important part of the design.
- Achieved through bold fonts, bright colors, or larger size.
- Example: “Sign Up” button highlighted in a distinct color.

4. Consistency

- Maintains uniformity across pages (fonts, colors, layouts).
- Builds user trust and improves navigation.
- Example: Same navigation bar style on every page.

5. White Space (Negative Space)

- Empty areas around elements that prevent clutter.
- Improves readability and focus.
- Example: Padding around text blocks.

6. Color Palette

- Harmonious use of colors to create mood and identity.
- Should align with brand and improve accessibility.
- Example: Blue for trust, green for growth, red for urgency.

7. Typography

- Choice of fonts affects readability and tone.
- Limit to 2–3 fonts for clarity.
- Example: Sans-serif for body text, serif for headings.

8. Movement & Flow

- Guides the eye through the design using layout patterns.
- Common patterns: Z-pattern (Western reading style) and F-pattern (scanning text online).

9. Unity

- Ensures all elements feel connected and part of one design.
- Achieved through consistent style, alignment, and theme.

10. Accessibility

- Designs must be usable by all, including people with disabilities.
- Example: High contrast text, alt text for images, responsive layouts.



7. Explain desirable features of a graphics design packages.

Desirable Features of a Graphics Design Package

A good graphics design package should provide tools and functions that make designing easy, efficient, and professional.

Key Features

1. User-Friendly Interface

- Simple menus, toolbars, and icons.
- Easy navigation for beginners and professionals.

2. Support for Multiple File Formats

- Ability to import/export in formats like JPEG, PNG, GIF, SVG, PDF, PSD, AI.
- Ensures compatibility across platforms.

3. Vector and Raster Support

- Vector tools for logos and scalable graphics.
- Raster tools for photo editing and detailed images.

4. Layer Management

- Organize elements in layers for flexible editing.
- Features like lock, hide, merge, and group layers.

5. Color Management

- Wide color palettes, gradients, and transparency options.
- Support for CMYK and RGB modes for print and digital.

6. Text and Typography Tools

- Variety of fonts, text effects, alignment, and spacing options.
- Essential for posters, ads, and web design.

7. Drawing and Editing Tools

- Brushes, pens, shapes, filters, and effects.
- Precision tools like grids, rulers, and snapping.

8. 3D and Animation Support (Advanced)

- Tools for 3D modeling, rendering, and motion graphics.
- Useful in multimedia and game design.

9. Compatibility and Integration

- Works with other software (e.g., Photoshop ↔ Illustrator ↔ AutoCAD).
- Cloud storage and collaboration features.

10. Performance and Stability

- Fast rendering, minimal crashes, and efficient memory usage.
- Handles large files smoothly.

8. What is computer animation? Explain about key-frame and morphing in detail.

Computer Animation is the art of creating moving images using computer graphics. It is essentially a digital version of the "flip-book" effect—by displaying a sequence of still images (frames) very quickly, the human eye perceives smooth, continuous motion.

Key-Frame Animation

- Definition: A technique where the animator defines important frames (called key frames) at specific points in time.
- The computer automatically generates the intermediate frames (called in-betweens) between key frames.

Process:

1. Animator sets starting and ending positions of an object.
 2. Software interpolates motion between them.
- Example: Moving a ball from left to right — animator sets ball at left (frame 1) and right (frame 20). The computer fills frames 2–19.
 - Advantages: Saves time, smooth motion, widely used in animation software like Flash, Blender, Maya.

Morphing

- Definition: Morphing is a special animation technique where one image gradually transforms into another.
- It uses interpolation of shapes and colors to create smooth transitions.
- Process:
 1. Define source image and target image.
 2. Identify key points (features like eyes, mouth, outline).
 3. Computer generates intermediate frames blending both images.
- **Example:** A human face morphing into an animal face in movies.
- **Applications:** Film special effects, advertising, creative animations.
- **Advantages:** Creates visually appealing transformations, used for dramatic effects.

9. Write short notes on.

a) CAD and CAM

CAD (Computer-Aided Design) aur **CAM (Computer-Aided Manufacturing)** dono engineering aur manufacturing ke backbone hain.

- **CAD:** Iska use engineers aur architects 3D models aur technical drawings banane ke liye karte hain. Ye manual drafting ko replace karta hai aur accuracy badhata hai.
 - *Examples:* AutoCAD, SolidWorks.
- **CAM:** Jab design (CAD) ban jata hai, toh CAM software us design ko "Machine Instructions" (G-codes) mein badal deta hai taaki CNC machines se physical parts banaye ja sakein.
- **Integration:** CAD/CAM ka combo product development ko fast banata hai aur errors ko kam karta hai.

b) 3D Transformation

3D Transformation ka matlab hai kisi 3D object ki position, size ya orientation ko x, y, z axes par badalna.

Main Types:

1. **Translation:** Object ko ek jagah se dusri jagah move karna. Isme (t_x, t_y, t_z) values add ki jati hain.
2. **Scaling:** Object ka size bada ya chhota karna. Isme (s_x, s_y, s_z) factors se multiply kiya jata hai.
3. **Rotation:** Object ko kisi axis (X, Y, Z) ke round ghumana. Isme sine aur cosine angles ka use hota hai.
4. **Shearing:** Object ki shape ko distort ya tircha karna.

c) Shading Models

Shading model ye determine karta hai ki light kisi object ki surface par kaise behave karegi taaki wo 3D lage.

- **Flat Shading:** Sabse simple model jahan poore polygon par ek hi color apply hota hai. Ye thoda "blocky" dikhta hai.
- **Gouraud Shading:** Isme polygon ke corners (vertices) par light calculate hoti hai aur beech mein color blend kiya jata hai. Ye smooth dikhta hai par "shiny spots" (highlights) acche nahi aate.
- **Phong Shading:** Ye har pixel par light calculate karta hai. Isse curved objects ekdam realistic aur shiny (specular highlights) dikhte hain.

d) Graphics File Formats

Graphics formats ko do main categories mein baata gaya hai:

1. **Raster Formats (Pixel-based):**
 - **JPEG:** Photography ke liye best, compressed hota hai.
 - **PNG:** Background transparent rakhne ke liye best, quality loss nahi hoti.
 - **GIF:** Simple animations aur limited colors ke liye.
2. **Vector Formats (Math-based):**
 - **SVG:** Web graphics ke liye best, kitna bhi zoom karo fat-ta nahi hai.
 - **EPS/PDF:** Printing aur professional designing ke liye use hote hain.

REGULAR/BACK EXAM-2079/2080, CHAITRA/BAISHAKH

1. Differentiate between raster and vector display technology.

Raster vs. Vector Display Technology

S.No	Feature	Raster Scan Display	Vector (Random Scan) Display
1	Basic Unit	Uses a grid of tiny dots called pixels .	Uses mathematical line segments .
2	Beam Path	The electron beam sweeps the entire screen, row by row.	The beam moves only to parts where a line is drawn.
3	Image Storage	Uses a Frame Buffer to store the color of every pixel.	Uses a Display List to store commands (e.g., "Draw Line").
4	Resolution	Limited by pixel density; can look jagged (pixelated).	Very high; lines are perfectly smooth and sharp.
5	Realism	Excellent for photos, shading, and complex colors.	Limited to "wireframe" models and simple line art.
6	Refresh Rate	Constant, regardless of how complex the image is.	Slows down as more lines are added, causing flicker.
7	Memory	High memory required to store every pixel's data.	Low memory required (only stores line coordinates).
8	Common Use	Modern TVs, Monitors, and Smartphones.	Oscilloscopes, Radar, and early CAD systems.

2. Digitize line joining points (2, 3) and (7, 9) Using DDA.

digitize a line using the **DDA (Digital Differential Analyzer)** algorithm, we follow a step-by-step incremental approach. Here is the detailed solution for joining points $(2, 3)$ and $(7, 9)$.

Step 1: Identify coordinates

- **Starting Point** $(x_1, y_1): (2, 3)$
- **Ending Point** $(x_2, y_2): (7, 9)$

Step 2: Calculate Δx and Δy

- $\Delta x = x_2 - x_1 = 7 - 2 = 5$
- $\Delta y = y_2 - y_1 = 9 - 3 = 6$

Step 3: Determine the Number of Steps

The number of steps is the maximum of the absolute values of Δx and Δy .

- $Steps = \max(|\Delta x|, |\Delta y|) = \max(5, 6) = 6$

Step 4: Calculate the Increment for each step

- $x_{inc} = \frac{\Delta x}{Steps} = \frac{5}{6} \approx 0.833$
- $y_{inc} = \frac{\Delta y}{Steps} = \frac{6}{6} = 1.0$

Step 5: Iteration Table

We start at $(2, 3)$ and repeatedly add the increments. To get the actual pixel on the screen, we round the values to the nearest integer.

k	x	y	Round(x)	Round(y)	Pixel Plot (x,y)
0	2.00	3.00	2	3	$(2, 3)$
1	2.83	4.00	3	4	$(3, 4)$
2	3.67	5.00	4	5	$(4, 5)$
3	4.50	6.00	5	6	$(5, 6)$
4	5.33	7.00	5	7	$(5, 7)$
5	6.17	8.00	6	8	$(6, 8)$
6	7.00	9.00	7	9	$(7, 9)$

(Note: In step $k=3$, 4.5 is usually rounded up to 5 in standard graphics algorithms to maintain symmetry.)

3. Explain various 2D transformations.

D Transformations in Computer Graphics

2D transformations are mathematical operations used to change the **position, size, orientation, or shape** of objects in a two-dimensional plane. They are represented using **matrix operations**.

1. Translation

- **Definition:** Shifts an object from one position to another.
- **Formula:**

$$x' = x + t_x, \quad y' = y + t_y$$

- **Example:** Moving a square 5 units right and 3 units up.

2. Scaling

- **Definition:** Changes the size of an object (enlarges or shrinks).
- **Formula:**

$$x' = x \cdot s_x, \quad y' = y \cdot s_y$$

- **Example:** Doubling the width and halving the height of a rectangle.

3. Rotation

- **Definition:** Rotates an object about the origin (or a point).
- **Formula (about origin):**

$$x' = x \cos \theta - y \sin \theta, \quad y' = x \sin \theta + y \cos \theta$$

- **Example:** Rotating a triangle 90° counter-clockwise.

4. Reflection

- **Definition:** Produces a mirror image of an object across a line (axis).
- **Types:**
 - Across **x-axis**: $(x, y) \rightarrow (x, -y)$
 - Across **y-axis**: $(x, y) \rightarrow (-x, y)$
 - Across **y = x**: $(x, y) \rightarrow (y, x)$

5. Shearing

- **Definition:** Distorts the shape of an object by shifting it in x or y direction.
- **Formula:**

$$x' = x + sh_x \cdot y, \quad y' = y + sh_y \cdot x$$

- **Example:** A rectangle becomes a parallelogram.

4. Define projection. Explain the types of projection with example.

Projection in computer graphics is the process of mapping 3D objects onto a 2D plane.

It helps visualize 3D scenes on flat surfaces like monitors or paper. Main types are parallel (true size) and perspective (realistic depth).

Types of Projection

Projections are broadly divided into two main categories: **Parallel Projection** and **Perspective Projection**.

1. Parallel Projection

In this type, the distance from the center of projection to the project plane is infinite. This means the projection lines (projectors) are parallel to each other.

- **Key Feature:** It preserves the relative proportions of the object. The size of the object does not change even if it moves further away.
- **Example:** Engineering blueprints or architectural floor plans (CAD).
- **Sub-types:**
 - **Orthographic:** The projectors are perpendicular to the projection plane (e.g., Top, Front, and Side views).
 - **Oblique:** The projectors are NOT perpendicular to the projection plane.

2. Perspective Projection

This type mimics how the human eye or a camera sees the world. The projectors converge at a single point called the **Center of Projection (COP)** or vanishing point.

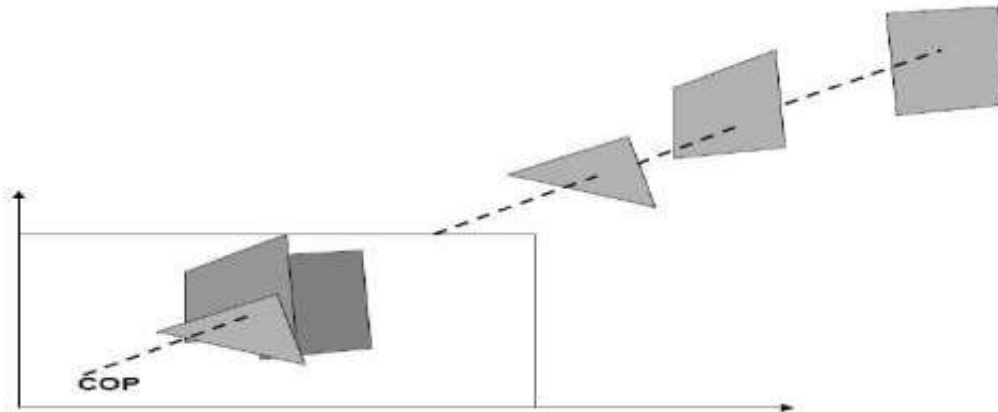
- **Key Feature: Foreshortening** occurs—objects that are farther away appear smaller than objects that are close.
- **Example:** 3D video games (like GTA or Minecraft), movies, and realistic landscape paintings.
- **Vanishing Points:** Parallel lines that are not parallel to the projection plane appear to meet at a single point.

5. Explain surface detection technique in brief.

Surface Detection Technique (Hidden Surface Detection) – Brief Explanation

Surface detection technique is used in **computer graphics** to identify which surfaces of a 3D object are **visible to the viewer** and which are **hidden** behind other surfaces.

Hidden surfaces are not displayed to make the image realistic and efficient.



Purpose

- Remove invisible (hidden) surfaces
- Improve realism of 3D images
- Reduce unnecessary drawing operations

Common Surface Detection Techniques (Brief)

1. Back Face Detection

- Surfaces facing away from the viewer are not drawn.
- Uses surface normal direction.

2. Z-Buffer (Depth Buffer) Method

- Stores depth (z-value) of each pixel.
- Pixel with smaller depth (closer to viewer) is displayed.

3. Painter's Algorithm

- Surfaces are drawn from farthest to nearest.
- Nearer surfaces overwrite farther ones.

4. Scan-Line Method

- Compares depth of surfaces line by line.
- Displays only the closest surface.

5. Define shading. Explain its types

Shading in computer graphics is the process of adjusting surface colors and brightness based on light. It makes objects look realistic by simulating how light interacts with surfaces. Different shading models are used to achieve smoothness, depth, and highlights.

Types of Shading

1. Flat Shading

- One single color intensity is applied to the **entire surface**.
- Lighting is calculated **once per polygon**.
- **Fast but looks rough** (edges are clearly visible).

Use: Simple objects, low-performance systems.

2. Gouraud Shading

- Light intensity is calculated at **each vertex**.
- Colors are smoothly interpolated across the surface.
- **Better than flat shading**, but highlights may be weak.

Use: Smooth surfaces with moderate performance.

3. Phong Shading

- Surface normals are interpolated **per pixel**.
- Lighting is calculated **for every pixel**.
- Produces **very smooth and realistic highlights**.

Use: High-quality realistic graphics.

7. Explain about morphing.

Morphing in computer graphics is a technique of smoothly transforming one image into another. It uses interpolation of shapes, colors, and features to generate intermediate frames. Commonly applied in movies, animations, and multimedia for dramatic visual effects.

Explanation

- **Process:**
 1. Select a source image and a target image.
 2. Identify key feature points (e.g., eyes, mouth, outline).
 3. The computer generates intermediate frames by blending geometry and color.
- **Result:** A continuous animation where the first image gradually changes into the second.
- **Applications:**
 - Film special effects (face morphing).
 - Advertising and creative animations.
 - Multimedia presentations.

8. Write down the purposes and features of graphic packages

Purposes of Graphic Packages

- To create and edit images (logos, posters, illustrations, 3D models).
- To enhance photos with filters, effects, and corrections.
- To design layouts for publishing, advertising, and web content.

- To support CAD/CAM in engineering and architecture.
- To produce animations and multimedia for films, games, and presentations.

Features of Graphic Packages

1. User-Friendly Interface → Easy menus, toolbars, drag-and-drop.
2. Multiple File Format Support → JPEG, PNG, GIF, SVG, PDF, PSD, AI.
3. Vector & Raster Tools → Scalable drawings + detailed photo editing.
4. Layer Management → Organize, lock, merge, and edit layers.
5. Color Management → Palettes, gradients, transparency, CMYK/RGB modes.
6. Text & Typography Tools → Fonts, alignment, spacing, effects.
7. Drawing & Editing Tools → Brushes, pens, shapes, filters, rulers.
8. 3D & Animation Support → Modeling, rendering, motion graphics.
9. Integration & Compatibility → Works with other apps, cloud storage.
10. Performance & Stability → Fast rendering, handles large files smoothly.

.9 Write short notes on:

i) Polygon Table

A **Polygon Table** is a data structure used to represent the geometric description of a 3D object. Since objects are made up of many connected polygons, tables are used to store this information efficiently to avoid redundancy.

It generally consists of two types of tables:

1. **Vertex Table:** Stores the coordinates (x, y, z) of every unique vertex in the object.
2. **Edge Table:** Lists the starting and ending vertex indices for each edge.
3. **Polygon-Surface Table:** Lists which edges or vertices make up each individual face (polygon).

Purpose: This organized structure helps the computer quickly calculate transformations (like rotation) and identify which surfaces are visible or hidden.

ii) Animation Sequence

An **Animation Sequence** refers to the specific order of steps and frames required to create the illusion of motion. In computer graphics, this involves a carefully planned workflow:

1. **Storyboard:** A series of rough sketches showing the main action.
2. **Object Definition:** Creating the 3D models or 2D characters.
3. **Keyframe Specification:** Defining the "main" frames at important points in time (e.g., the start and end of a jump).
4. **In-betweening (Tweening):** The computer automatically generates the frames between keyframes to make the motion smooth.

Purpose: It ensures that the movement of objects is fluid, logically timed, and follows the laws of physics or the creative vision of the animator.

iii) Graphics File Format

A **Graphics File Format** is a standardized way of organizing and storing digital images. Choosing the right format depends on whether you need high quality, small file size, or transparency.

There are two main categories:

1. **Raster Formats:** Store images as a grid of pixels.
 - *JPEG*: Great for photos (compressed).
 - *PNG*: Supports transparency (lossless).
 - *GIF*: Supports simple animations.
2. **Vector Formats:** Store images as mathematical paths.
 - *SVG*: Perfect for logos and icons (can be resized infinitely without losing quality).
 - *PDF*: Used for documents and printing.

Purpose: Different formats allow for optimized performance, whether the image is meant for a website, a high-quality print, or a professional 3D design.

REGULAR/BACK EXAM-2079, BHADRA /ASHWIN

1. Define Computer Graphics. Explain applications of Computer Graphics.

Definition:

Computer Graphics is the branch of computer science that deals with **creating, storing, manipulating, and displaying images and visual information** using computers.

It converts data into **pictures, diagrams, and animations** for better understanding

.. Explain applications of Computer Graphics.

1. Computer-Aided Design (CAD)

Used by engineers and architects to create precise 1:1 scale models of buildings, mechanical parts, and electrical circuits. It allows for testing and "virtual prototyping" before actual manufacturing begins.

2. Entertainment (Movies and Games)

This is the most visible application. It includes **CGI (Computer Generated Imagery)** in films and real-time rendering in video games to create realistic characters, environments, and special effects.

3. Medical Imaging

Graphics technology is vital for visualizing the human body. Techniques like **MRI, CT Scans, and Ultrasound** convert raw sensor data into 3D visual models that doctors use for diagnosis and surgery planning.

4. Education and Training

Computer-generated models are used in simulators, such as **Flight Simulators** for pilot training. It provides a safe, cost-effective environment to practice complex tasks.

5. Visualization (Data and Scientific)

Scientists and data analysts use graphics to represent complex sets of numbers as charts, graphs, or 3D maps. This helps in identifying patterns in weather systems, stock markets, or molecular structures.

6. Graphical User Interface (GUI)

Every time you use a smartphone or a computer, you are interacting with computer graphics. Icons, windows, buttons, and menus are all graphical elements that make devices user-friendly.

7. Digital Art and Advertising

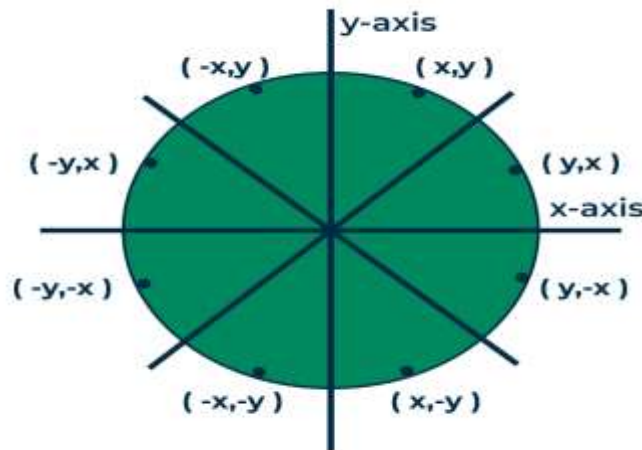
Artists use graphic tablets and software (like Photoshop or Illustrator) to create digital paintings, logos, and promotional banners for marketing.

2. Derive midpoint circle drawing algorithm.

Midpoint Circle Drawing Algorithm – Derivation

The **midpoint circle drawing algorithm** is an efficient algorithm used in **computer graphics** to draw a circle using only **integer calculations**.

It is based on the **symmetry of a circle** and the **midpoint decision method**



Midpoint circle drawing algorithm

1. Circle Equation

The general equation of a circle centered at origin with radius r is;

$$x^2 + y^2 = r^2$$

2. Idea of Midpoint

- Start from point $(0, r)$.
- At each step, move one pixel in x direction and decide whether move in y direction too.
- Decision is based on the midpoint between two possible pixels.

3. Decision Parameter

Let the midpoint be $(x+1, y-\frac{1}{2})$.

Substitute in circle equation:

$$p = (x+1)^2 + (y-\frac{1}{2})^2 - r^2$$

4. Initial Condition

- $1 \times p_k < 0 \Rightarrow p_k < 7 - r$

$$p_{k+1} = p_k + 2x_k + 3$$

$$p_{k+1} = p_k + 2y_k + 5$$

6. Symmetry

Since circle is symmetric, only one octant is calculated and other points are mirrored

3. Define projection. Write down differences between parallel and perspective projection with figure.

Definition: Projection is the process of mapping a 3D object from its original 3D coordinate system (x, y, z) onto a 2D plane (the screen). It is essential because computer screens are 2D surfaces, but the scenes we often create are 3D.

S.N.	Feature	Parallel Projection	Perspective Projection
1	Projection lines	Parallel to each other	Converge to a single point (vanishing point)
2	Object size	Remains constant	Decreases as distance increases
3	Depth perception	No depth effect	Shows depth (realistic)
4	Realism	Less realistic	More realistic
5	Distortion	Shape and angles preserved	Shape may get distorted
6	Use/Application	Engineering drawings, CAD	Movies, animation, games
7	Complexity	Simple to implement	More complex to implement
8	Example	Technical diagrams, blueprints	3D visualizations, 3D games

4. What is animation? Explain steps of animation & sequence.

definition: Animation is a technique where images or objects are displayed in a rapid sequence to create the illusion of movement. Each image is slightly different from the previous one, and when played at high speed, it looks like motion

Steps of Animation (Sequence)

To create a good animation, the following steps are followed:

1. **Storyboard Layout:** This is the first step of animation. Rough sketches are made to show the entire story or action frame-by-frame. It gives an idea of how the final animation will look.
2. **Object Definition:** In this step, the objects that will move in the animation are defined. The shape, size, and geometry (like polygons or lines) of the objects are stored in the computer.
3. **Keyframe Specification:** Keyframes are the main frames that show the starting and ending positions of an action. For example, if a ball is jumping, its starting point and the highest point are the keyframes.
4. **Generation of In-between Frames:** This is also called Tweening. The computer automatically generates all the frames between two keyframes so that the motion looks smooth and natural.
5. **Rendering:** This is the final step where the animation is given a final touch. Light, color, shadows, and textures are added to make the scene look realistic.

Animation Sequence (Important Points)

- **Frame Rate:** For a smooth animation, there should be at least 24 frames per second (fps).
- **Timing:** The speed at which frames play decides how realistic the movement looks.
- **Looping:** If an action needs to repeat (like walking), the sequence is set to "loop."

. 5.Explain graphics file formats. What are the principle of web graphics design?

Graphics File Formats

Definition:

Graphics file format defines **how images are stored, compressed, and displayed** in a computer.

Different formats store images differently, depending on whether the image is **raster (bitmap)** or **vector**.

Types of Graphics File Formats:

1. Raster (Bitmap) Formats

- Image is made of pixels.
- Examples: **JPEG, PNG, BMP, GIF**
- Advantages: Realistic images, supports photos.
- Disadvantages: Enlarging reduces quality (pixelation).

2. Vector Formats

- Image is made of **lines, curves, and shapes**.
- Examples: **SVG, EPS, PDF**
- Advantages: Can scale without losing quality, smaller file size.
- Disadvantages: Not suitable for detailed photos.

Uses:

- Storing, sharing, and displaying images on web or print.
- Multimedia and animation applications.

Principles of Web Graphics Design

1. Simplicity

- Keep design clean, avoid clutter.
- Focus on important elements.

2. Consistency

- Use consistent colors, fonts, and style across pages.

3. Readability

- Text should be easy to read; background should not distract.

4. Visual Hierarchy

- Highlight important elements using size, color, or position.

5. Optimized Images

- Use correct format and size to reduce page loading time.

6. Accessibility

- Ensure graphics are visible and understandable for everyone, including visually impaired.

7. Responsive Design

- Graphics should adapt to different screen sizes (desktop, tablet, mobile).

8. Attractiveness

- Use color, balance, and alignment to make the site visually appealing.

6. Why GUI is popular than CUI? What are the principle of interactive user design? Explain three of them.

Why GUI is more popular than CUI?

GUI (Graphical User Interface) is much more popular than CUI (Character User Interface) because of the following reasons:

1. **User Friendly:** In GUI, we use visual elements like icons, menus, and windows. We don't need to memorize complex commands like in CUI, making it easy for beginners.
2. **Mouse Support:** GUI supports pointing devices like a mouse or touch screen. In CUI, we can only use the keyboard to type commands, which is slower.
3. **Multitasking:** GUI allows us to open and work on multiple windows at the same time. In CUI, a user generally works on one task or command at a time.
4. **Visual Feedback:** In GUI, we get immediate visual feedback (like a progress bar or a folder opening). In CUI, we only see text, which can be confusing if a command fails.
5. **Graphics Support:** GUI can display high-quality images, videos, and animations, which is not possible in a text-based CUI.

Principles of Interactive User Design

Interactive design is about how a user communicates with a computer. To make a design effective, certain principles must be followed.

Three important principles are:

1. **Consistency:** All parts of the interface should look and behave the same way. For example, if the "Save" button is in the top-left corner of one screen, it should be in the same place on other screens. This helps users learn the system faster.
2. **Feedback:** The system should always provide a response for every action. For example, when you click a button, it should change color or show a "Loading" sign. This tells the user that the computer has received their request.
3. **User Control:** The user should feel in control of the interface. This means there should be an easy way to "Undo" or "Cancel" an action if a mistake is made. It encourages users to explore the software without fear of breaking something.

7. What are desirable features of graphic design package? Explain morphing technique.

Desirable Features of Graphic Design Package

Definition

Graphic design packages are software tools used to create, edit, and manage visual content.

They offer features for drawing, coloring, layering, and exporting designs.

Good packages improve creativity, productivity, and output quality.

Desirable Features:

-  User-friendly interface – Easy navigation, drag-and-drop tools.
-  Multi-format support – JPEG, PNG, SVG, PSD, PDF, etc.
-  Layer management – Add, lock, merge, and edit layers.
-  Drawing tools – Brushes, pens, shapes, gradients.
-  Text tools – Font styles, alignment, effects.
-  Color control – Palettes, transparency, CMYK/RGB modes.
-  Smart guides & snapping – Helps align objects precisely.
-  Plugin support – Extend functionality with add-ons.
-  Templates & presets – Speeds up design work.
-  Performance & stability – Handles large files smoothly.

Morphing Technique

Definition (3 lines):

Morphing is a technique of smoothly transforming one image into another.

It blends shapes and colors to create intermediate frames.

Used in animation, films, and multimedia for visual transitions.

Explanation:

- Start with source image and target image.
- Identify key feature points (e.g., eyes, outline).
- Use interpolation to generate intermediate frames.
- Final result: smooth transition from one image to another.
-  Example: Face morphing in movies or logo transitions in ads.

8. Write the principle and characteristics of raster and vector display technology.

Explain the working principle of LCD monitor.

Raster and Vector Display Technology

Raster Scan Display:

- **Principle:** The electron beam sweeps across the screen, one row at a time (from top to bottom). As the beam moves, it turns on and off to create a pattern of illuminated dots called pixels.
- **Characteristics:**
 1. Uses a "Refresh Buffer" (Frame Buffer) to store pixel intensity.
 2. Can display realistic images with different shades and colors.
 3. Produces "aliasing" (jagged edges) on diagonal lines.
 4. Used in modern monitors and TVs.

Vector Scan Display (Random Scan):

- **Principle:** The electron beam is directed only to the parts of the screen where a picture is to be drawn. It draws lines directly from one point to another, just like a pen on paper.
- **Characteristics:**
 1. Highest resolution for line drawings.
 2. No "aliasing" (diagonal lines are smooth).
 3. Cannot display complex shaded images or realistic photos.
 4. Used in early radar systems and oscilloscope displays.

Working Principle of LCD Monitor

1. **Backlight Source**
 - A fluorescent lamp or LED provides uniform white light behind the screen.
2. **Polarizing Filters**
 - Two polarizers are placed at right angles.
 - Light passes only when liquid crystals align correctly.
3. **Liquid Crystal Layer**
 - Liquid crystals twist or align when voltage is applied.
 - This changes the angle of light passing through.
4. **Color Filters (RGB)**
 - Each pixel is divided into red, green, and blue sub-pixels.
 - By controlling intensity, millions of colors are produced.
5. **Matrix Control (TFT Technology)**
 - Thin Film Transistors (TFT) control each pixel individually.
 - Ensures sharp images and fast response.

9. Write short notes on.

a) Phong Shading

Phong shading is a sophisticated interpolation technique used in 3D computer graphics to create realistic lighting and highlights on a surface.

- **Principle:** Unlike simpler methods (like Gouraud shading) that calculate light at vertices, Phong shading calculates the surface **normal vector** at every single pixel.
- **Process:** It interpolates the normal vectors across the surface of a polygon and applies the lighting model (ambient, diffuse, and specular) at each pixel.
- **Key Feature:** Its main advantage is the ability to produce accurate **specular highlights** (the shiny "hot spot" on a glossy object).
- **Drawback:** It is computationally more expensive than Gouraud shading because it requires more calculations per pixel.

b) 2D Transformation

2D Transformation refers to applying mathematical operations to a 2D object to change its position, size, or orientation in a coordinate system.

The three basic types of 2D transformations are:

1. **Translation:** Moving an object from one position (x, y) to another (x', y') by adding translation distances (tx, ty) .
 - $x' = x + tx$
 - $y' = y + ty$
2. **Rotation:** Turning an object around a fixed point (usually the origin) by a specific angle θ .
3. **Scaling:** Changing the size of an object. It involves multiplying the coordinates by scaling factors (sx, sy) .
 - $x' = x \cdot sx$
 - $y' = y \cdot sy$

[Image showing 2D Translation, Rotation, and Scaling of a triangle]

c) Resolution

Resolution is a measure used to describe the clarity and sharpness of an image or a display screen.

- **Definition:** It is usually defined by the number of **pixels** (dots) per unit area or the total number of pixels in the width and height.
- **Types:**
 1. **Screen Resolution:** Expressed as (width $\times$ height), such as 1920×1080 . A higher number of pixels means a sharper display.
 2. **Image Resolution:** Often measured in **PPI** (Pixels Per Inch). Higher PPI results in more detail when printed or zoomed.
- **Aspect Ratio:** This is the proportional relationship between the width and the height (e.g., 16:9 for widescreen).
- **Importance:** Resolution determines the "graininess" of a graphic. If the resolution is too low for the screen size, the image will look "pixelated" (blurry or blocky).

REGULAR/BACK EXAM-2078/2079, CHAITRA/BAISHAKH

1. Define computer graphics. Write down the major applications of computer graphics

Definition.

Computer graphics is the branch of computer science that deals with generating, manipulating, and displaying visual content using computers. It involves converting data into images, drawings, and animations for better understanding and communication. It is widely used in design, entertainment, education, and scientific visualization.

Major Applications of Computer Graphics

Computer graphics are used in almost every field today. Here are the most common applications:

- 1. Computer-Aided Design (CAD):** Used by engineers and architects to design buildings, automobiles, and electronic circuits. It allows for precise measurements and 3D modeling.
- 2. Presentation Graphics:** Used to create reports, bar charts, pie charts, and graphs to summarize data for business or educational purposes. It makes data easy to understand at a glance.
- 3. Entertainment (Movies and Games):** This is the most visible application. It includes creating CGI (Computer Generated Imagery) for movies, special effects, and high-definition 3D games.
- 4. Education and Training:** Used to create simulators (like flight simulators for pilots) or educational models (like a 3D model of a human heart) to help students learn complex topics safely and easily.
- 5. Medical Imaging:** Doctors use computer graphics for CT scans, MRIs, and X-rays to see inside the human body without surgery. It helps in planning surgeries and diagnosing diseases.
- 6. Image Processing:** Involves modifying existing images to improve their quality, remove noise, or apply filters. This is commonly seen in photo editing apps like Photoshop.
- 7. Graphical User Interface (GUI):** The icons, windows, and buttons you see on your computer or phone screen are all created using computer graphics to make the device easy to use.

2. Explain Bresenham's line drawing algorithm.].

Bresenham's Line Drawing Algorithm

Bresenham's algorithm is an efficient method to draw straight lines using only integer calculations. It determines the nearest pixel to approximate the line without using floating-point arithmetic.

Key Formula:

Decision Parameter:

$$P_0 = 2\Delta y - \Delta x$$

$$\Delta x = x_2 - x_1$$

$$\Delta y = y_2 - y_1$$

Steps of the Algorithm:

- ① Start: x_1, y_1 to x_2, y_2
- ② Calculate initial $P_0 = 2\Delta y - \Delta x$
- ③ Plot (x_1, y_1)
- ④ For each x till x_2 :

1. If $P_k < 0$, then $x_{k+1} = x_k + 1, y_{k+1} = y_k$

2. Else $P_k \geq 0$, then $x_{k+1} = x_k + 1, y_{k+1} = y_k + 1$

$$P_{k+1} = P_k + 2\Delta y - 2\Delta x$$

- ⑤ Repeat until $x = x_2$

Plots the line efficiently by choosing nearest pixels.



Principles of Web Graphics Design

3. Write down principles of web graphics design.

1. Clarity & Simplicity

- Design should be clean and easy to understand.
- Avoid clutter and focus on key content.

2. Consistency

- Use uniform colors, fonts, and layout across all pages.
- Builds trust and improves navigation.

3. Responsive Design

- Graphics should adapt to different screen sizes (mobile, tablet, desktop).
- Ensures usability across devices.

4. **Fast Loading**

- Optimize images and graphics to reduce file size.
- Improves performance and SEO.

5. **Visual Hierarchy**

- Use size, color, and placement to guide user attention.
- Important elements should stand out.

6. **Accessibility**

- Use alt text, readable fonts, and high contrast.
- Makes content usable for all users, including those with disabilities.

7. **Brand Alignment**

- Graphics should reflect the brand's tone, style, and identity.
- Helps in recognition and trust.

8. **User Engagement**

- Use interactive elements (hover effects, animations) wisely.
- Enhances experience without distraction.

4. **Explain surface detection techniques.**

Surface Detection Techniques

1. Back-Face Detection (Object-Space Method)

- Checks if a polygon is facing away from the viewer.
- If the surface normal points away, it's hidden.
- Fast and simple for convex objects.

2. Z-Buffer Method (Image-Space Method)

- Stores depth (Z) value for each pixel.
- Compares depth of incoming pixel with stored value.
- Keeps the closest pixel, discards others.
- Used in real-time rendering (e.g., OpenGL).

3. Scan-Line Method

- Processes one horizontal line (scan-line) at a time.
- Tracks overlapping surfaces and depth per line.
- Efficient for scenes with few overlapping objects.

4. Painter's Algorithm

- Sorts surfaces from farthest to nearest.
- Draws them in order → nearer surfaces overwrite farther ones.
- May fail with cyclic overlaps (needs splitting).

5. BSP Trees (Binary Space Partitioning)

- Divides space recursively into front and back halves.
- Helps in sorting surfaces efficiently.
- Used in game engines and indoor scenes.

5. Difference Between Raster and Vector Display Technology

S.N.	Feature	Raster Display	Vector Display
		Made of pixels arranged in rows and columns	Made of lines, curves, and shapes using mathematical equations
		Depends on number of pixels; enlarging reduces quality	Resolution independent; can scale without loss of quality
		Suitable for realistic images, photos	Suitable for line drawings, diagrams, CAD
		Refreshes screen line by line (scan lines)	Electron beam directly draws lines along paths
		Slower for large images	Faster for simple objects
		Large file size for high resolution	Smaller file size
		LCD, LED monitors, TV, digital images	Oscilloscope, CAD displays
		Photography, videos, multimedia	Engineering graphics, technical drawings

6. What is projection? Differentiate between parallel and perspective projection

Projection in computer graphics is the process of mapping 3D objects onto a 2D plane. It helps visualize 3D scenes on flat surfaces like monitors or paper.

Two major types are parallel projection and perspective projection.

Parallel vs Perspective Projection

S.N.	Feature	Parallel Projection	Perspective Projection
1	Projection Lines	Parallel to each other	Converge to a single point (vanishing point)
2	Object Size	Remains constant	Decreases with distance
3	Depth Perception	No depth effect	Shows realistic depth
4	Realism	Less realistic	More realistic
5	Distortion	Shape and angles preserved	Shapes may get distorted
6	Usage	CAD, technical drawings	Movies, 3D visualization, games
7	Complexity	Simple	More complex
8	Example	Engineering diagrams, blueprints	3D animation, architectural rendering

7. Define shading model. Explain about Gouraud shading.

A shading model is a method used in computer graphics to determine how light interacts with surfaces. It calculates the color and brightness of pixels based on light sources, surface properties, and viewer position. Common shading models include Flat, Gouraud, and Phong shading.

Explain about Gouraud shading. Explanation:

- Developed by Henri Gouraud (1971).
- Works by calculating illumination at polygon vertices using the lighting model.
- These vertex intensities are then interpolated across the surface of the polygon.
- Produces smooth shading across surfaces, avoiding the sharp edges of flat shading.
- Efficient because lighting calculations are done only at vertices, not every pixel.

Steps:

1. Compute intensity at each vertex using a lighting model (e.g., Lambert's law).
2. Interpolate intensity values along polygon edges.
3. Interpolate across scan lines to fill the polygon.
4. Result: smooth gradient shading across the surface.

Advantages:

- Faster than Phong shading (less computation).
- Produces smooth surfaces compared to flat shading.

Limitations:

- Specular highlights may be missed if they don't fall on a vertex.
- Less realistic than Phong shading.

8. Explain various steps of animation.

Steps of Animation

1. **Concept / Idea**
 - Define the story, theme, and purpose of the animation.
 - Decide the style (2D, 3D, stop-motion, etc.).
2. **Storyboarding**
 - Rough sketches of scenes and transitions.
 - Acts like a blueprint for the animation sequence.
3. **Modeling / Asset Creation**
 - Create characters, objects, and backgrounds.
 - In 3D animation, models are built using software tools.
4. **Layout & Scene Setup**
 - Place models in the scene.
 - Define camera angles, lighting, and environment.
5. **Animation (Movement Definition)**
 - Add motion to characters and objects.
 - Use key frames (important poses) and in-between frames (smooth transitions).
6. **Texturing & Lighting**
 - Apply colors, textures, and materials to models.
 - Adjust lighting for realistic effects.
7. **Rendering**
 - Generate the final sequence of frames.
 - Convert 3D/2D data into viewable images or video.
8. **Post-Production**
 - Add sound, effects, and editing.
 - Final polish before release

9. Write short notes on:

i) 2D Rotation

2D Rotation is a transformation technique used to rotate an object about a fixed point (called the **Pivot Point**) by a specific angle θ in a 2D plane.

- **Pivot Point:** By default, objects are rotated about the origin $(0,0)$. However, they can be rotated about any arbitrary point.
- **Direction:** Usually, a positive angle rotates the object in a **counter-clockwise** direction, and a negative angle rotates it **clockwise**.
- **Mathematical Formula:** If a point (x, y) is rotated by angle θ to a new position (x', y') , the equations are:

$$x' = x \cos \theta - y \sin \theta$$

$$y' = x \sin \theta + y \cos \theta$$

ii) Morphing

Morphing is a special effect in animation that changes (or "metamorphoses") one image into another through a seamless, fluid transition.

- **The Process:** It involves two main steps:
 1. **Warping:** The shapes of the two images are distorted to align with each other based on "control points" (like the eyes or mouth).
 2. **Cross-Dissolving:** The first image gradually fades out while the second image gradually fades in.
- **Applications:** It is widely used in movies (e.g., a person turning into a werewolf), music videos, and forensic science to show age progression.

iii) CAD vs CAM

While both are used in modern manufacturing, they serve different purposes in the production cycle.

Feature	CAD (Computer-Aided Design)	CAM (Computer-Aided Manufacturing)
Purpose	Used to design and create 2D/3D digital models of products.	Used to control machinery and automate the production process.
Focus	Focuses on geometry, dimensions, and material properties.	Focuses on tool paths, cutting speeds, and machining instructions.
Output	The output is a digital drawing or blueprint (e.g., .dwg or .step files).	The output is G-code or instructions that a CNC machine can understand.
Example	An architect designing a house layout.	A robot arm carving a car part out of metal based on a design.

REGULAR/BACK EXAM-2078, KARTIK/MANGSIR

1. Define computer graphics. Explain application of computer graphics. Differentiate between CAD and CAM.

Definition:

Computer Graphics is a sub-field of computer science that involves the generation, manipulation, and storage of visual images and geometric data using a computer. It transforms abstract data into visual representations, ranging from simple 2D line drawings to complex 3D animations and virtual reality.

Applications of Computer Graphics

Computer graphics are used across almost every modern industry. Key applications include:

- **Computer-Aided Design (CAD):** Used by engineers and architects to create precise 2D and 3D models of buildings, airplanes, and electronic circuits.
- **Entertainment (Movies and Gaming):** Creating CGI (Computer Generated Imagery), special effects, and immersive 3D environments for video games.
- **Medical Imaging:** Used for CT scans, MRIs, and X-rays to create visual models of the internal human body for diagnosis and surgery planning.
- **Education and Training:** Flight simulators for pilots and interactive 3D models for students (like exploring the solar system or human anatomy).
- **Presentation Graphics:** Turning complex business data into easy-to-read bar charts, pie charts, and graphs.
- **Image Processing:** Enhancing or modifying existing photos (like removing "noise" from a satellite image or using filters in Photoshop).
- **GUI (Graphical User Interface):** Everything you see on your screen—icons, menus, and windows—is a product of computer graphics.

Differentiate between CAD and CAM

While they are often used together in a process called **CAD/CAM**, they represent two different stages of production.

Feature	CAD (Computer-Aided Design)	CAM (Computer-Aided Manufacturing)
Primary Goal	To design and document the visual/geometric model of a product.	To manufacture the product using computer-controlled machinery.
Input	Ideas, sketches, and engineering requirements.	The digital model created in the CAD stage.
Output	2D/3D blueprints and technical drawings.	Instructions (G-code) for robots and CNC machines.
User	Architects, Product Designers, and Engineers.	Factory Managers and Machine Operators.
Action	Involves drawing, stretching, and scaling a virtual object.	Involves cutting, drilling, and milling a physical object.

2. What is touch pad? Explain its types in brief. Write differences between raster and vector display technology.

A touch pad is an input device that allows users to control the cursor by moving their finger across a flat surface. It senses finger movement and taps using pressure, capacitance, or optical sensors.

Commonly used in laptops as a substitute for a mouse.

Types of Touchpads

1. Capacitive Touchpad:

- **How it works:** This is the most common type. It uses the electrical properties of the human body to detect touch. The surface has a grid of sensors that track the change in capacitance when a finger touches it.
- **Feature:** It is very sensitive and supports multi-touch gestures (like pinching to zoom), but it usually won't work if you are wearing thick gloves.

2. Resistive Touchpad:

- **How it works:** It works based on physical pressure. It consists of two layers that touch each other when you press down.
- **Feature:** It is less common in modern laptops but can be used with a stylus or gloved hands. It is generally less precise than capacitive version

3. Differences: Raster vs. Vector Display

4. These are the two fundamental technologies used to produce images on a screen.

Feature	Raster Scan Display	Vector Scan Display
Basic Concept	Images are made of a grid of dots (pixels).	Images are made of mathematical lines and paths.
Beam Path	The electron beam sweeps across the entire screen, row by row.	The electron beam moves only to the parts where a line is drawn.
Resolution	Limited by the number of pixels. Zooming in makes it "pixelated."	Very high resolution. Lines remain smooth even when scaled.
Realism	Best for realistic photos, videos, and complex shading.	Only suitable for line drawings and wireframe models.
Memory	Uses a Frame Buffer to store every pixel's intensity.	Uses a Display File to store line-drawing commands.
Aliasing	Suffers from "jaggies" (jagged edges on diagonal lines).	No aliasing; diagonal lines are perfectly smooth.

3. Write Bresenham's line drawing algorithm. Digitize the line joining points (5, 5) and (10, 12) using BLA.

Bresenham's Line Drawing Algorithm

✓ Bresenham's Line Drawing Algorithm is an efficient method for rasterizing lines using only integer arithmetic. It determines which pixel is closest to the true line path at each step. It works for all slopes and octants by using symmetry.

Algorithm Steps

- Input: Two endpoints (x_0, y_0) and (x_1, y_1) .
- Calculate differences: $\Delta x = x_1 - x_0$, $\Delta y = y_1 - y_0$
- Initialize decision parameter: $p_0 = 2\Delta y - \Delta x$
- Start at point (x_0, y_0) .
- For each step in x:
 - If $p_k < 0$: next point = $(x_k + 1, y_k)$, update $p_{k+1} = p_k + 2\Delta y$
 - If $p_k \geq 0$: next point = $(x_k + 1, y_k + 1)$, update $p_{k+1} = p_k + 2\Delta y - 2\Delta x$
- Repeat until endpoint (x_1, y_1) is reached.

Digitization Example: Line from (5, 5) to (10, 12)

- Step 1: $(x_0, y_0) = (5, 5)$, $(x_1, y_1) = (10, 12)$
- $\Delta x = 5$, $\Delta y = 7$
- Since slope $m = \frac{\Delta y}{\Delta x} = \frac{7}{5} > 1$, we increment y each step and decide on x.
- Initial decision parameter: $p_0 = 2\Delta y - \Delta x = 14 - 5 = 9$
- Step-by-Step Points
 - Start: $(6, 5)$, $p_1 = 9 + 10 - 2 \cdot 5 = 9$
 - → next $(6, 7)$, $p_2 = 9 + 10 - 2 \cdot 5 = 4$
 - → next $(7, 8)$, $p_3 = 9 + 10 - 2 \cdot 5 = 9$
 - → next $(8, 9)$, $p_4 = 14 + 10 - 2 \cdot 5 = 9$
 - → next $(9, 10)$, $p_5 = 19 + 10 - 10 = 9$
 - End at $(10, 12)$, End at $(10, 12)$.

✓ Digitized Line Points: (5, 5), (6, 6), (6, 7), (7, 8), (8, 9), (9, 10), (10, 11).

4. a) Define projection Explain different types of projection.

Projection is the process of transforming a 3D object into a 2D plane (viewing surface) by mapping 3D coordinates to 2D coordinates. This simulates how the human eye or a camera captures a scene.

Types of Projection

1. Parallel Projection

- Projectors are parallel (no convergence).
- Preserves true shape and size for faces parallel to projection plane.
- **Types:**
 - **Orthographic:**
 - Projectors perpendicular to projection plane.
 - **Subtypes:**
 - **Top / Plan View** (XY-plane)
 - **Front / Elevation View** (XZ-plane)
 - **Side View** (YZ-plane)
 - **Axonometric:**
 - **Isometric:** All axes equally foreshortened (120° between axes)
 - **Dimetric:** Two axes equally foreshortened
 - **Trimetric:** All axes differently foreshortened
 - **Oblique:**
 - Projectors at an angle ($\neq 90^\circ$) to projection plane.
 - **Subtypes:**
 - **Cavalier:** Angle = 45° , foreshortening factor = 1
 - **Cabinet:** Angle = 63.4° , foreshortening factor = 0.5

2. Perspective Projection

- Projectors converge at a **Center of Projection (COP)**.
- Simulates realistic view (distant objects appear smaller).
- **Types** (based on vanishing points):
 - **One-point Perspective:** One principal face parallel to projection plane
 - **Two-point Perspective:** One axis parallel to projection plane
 - **Three-point Perspective:** No principal face parallel to projection plane

b) Explain different types of shading models.

Types of Shading Models

1. Flat Shading

- Single color for entire polygon.
- Uses surface normal of polygon centroid.
- Fast but produces **Mach band effect** (discontinuities between polygons).
- Suitable for distant objects or low-detail models.

2. Gouraud Shading (Intensity Interpolation)

- **Steps:**
 1. Compute vertex normals (average of adjacent face normals)
 2. Calculate lighting intensity at each vertex

3. Interpolate intensities across polygon scan lines

- Eliminates intensity discontinuities.
- **Limitations:** Does not handle specular highlights well, silhouette edges may appear polygonal.

3. Phong Shading (Normal-Vector Interpolation)

- More accurate than Gouraud.
- **Steps:**
 1. Interpolate vertex normals across polygon
 2. Compute lighting at **each pixel** using interpolated normal
- Better for specular highlights, smoother surfaces.
- **Disadvantage:** Computationally expensive.
- **5. Cel Shading (Non-Photorealistic)**
 - Creates cartoon-like appearance with discrete shading levels.
 - Uses step functions instead of smooth gradients. ..Outlines often added for emphasis.

5. What is animation? Explain about animation sequences.

Definition

Animation is the process of creating the **illusion of motion** and change by rapidly displaying a sequence of static images (frames) that differ slightly from one another. This illusion works due to a biological phenomenon called "**Persistence of Vision**," where the human eye retains an image for a fraction of a second, allowing multiple frames to blend into smooth movement.

Animation Sequences

The creation of a digital animation follows a structured sequence of steps to ensure the motion is realistic and the storytelling is clear:

1. **Storyboard Layout:** This is the planning phase where the entire action is sketched out scene-by-scene. It acts as a visual guide for the animators to understand camera angles, character positions, and the flow of the story.
2. **Object Definition:** In this step, the physical characteristics of the objects are created.
 - **Modeling:** Creating the 2D or 3D geometry (the "shape") of the character or object.
 - **Rigging:** Adding a digital skeleton to a 3D model so that animators can move the limbs and body parts.
3. **Keyframe Specification:** Keyframes are the "major" frames that define the start and end points of any smooth transition. The animator manually sets these frames to fix the primary positions of a character.
4. **Generation of In-betweens (Tweening):** This is the process of creating the frames that occur *between* the keyframes. In modern computer graphics, the software automatically calculates these intermediate frames to make the movement look fluid. This is known as **Tweening**.
5. **Rendering:** This is the final technical step where the computer processes all the mathematical data, lighting effects, textures, and shadows to produce the final video file.

6. Define Graphical User Interface (GUI). Explain about differ graphical interface items.

Definition

A GUI (pronounced "goo-ey") uses visual elements to represent the information and actions available to a user. It relies on a "point-and-click" system, usually managed by a mouse or a touchscreen. The most common model for a GUI is the **WIMP** system: **W**indows, **I**cons, **M**enus, and **P**ointers.

Graphical Interface Items

To make navigation easy, GUIs use several standard components:

1. Windows

A window is a rectangular area on the screen that displays a specific application or document. It allows for multitasking because you can open multiple windows at once, resize them, and switch between them.

2. Icons

Icons are small graphic symbols that represent files, folders, applications, or commands. Instead of typing a path like C:\Program Files\Chrome.exe, you simply double-click the Chrome icon.

3. Menus

Menus provide a list of choices or commands.

- **Pull-down Menus:** Usually found at the top of a window (e.g., File, Edit, View).
- **Context Menus:** Pop up when you "right-click" an item, showing actions specific to that item.

4. Pointers (Cursor)

A pointer is a symbol (usually an arrow) that moves on the screen as you move your mouse. It changes shape based on the task—for example, it becomes a "hand" icon when hovering over a link or an "I-beam" when clicking on text.

5. Toolbars and Ribbons

A toolbar is a row of icons, usually located just below the menu bar, that provides quick access to frequently used tools (like Save, Print, or Bold).

6. Dialog Boxes

These are small windows that pop up to provide information or ask the user for input. For example, when you try to close an unsaved file, a dialog box asks, *"Do you want to save changes?"*

7. Buttons and Checkboxes

- **Buttons:** Rectangular or rounded shapes that perform an action when clicked (e.g., "Submit" or "OK").
- **Checkboxes:** Small boxes that allow users to select one or more options from a list.

6. a) Explain the principles of web graphic design.

Web graphic design is not just about making a site look "pretty"; it is about creating a functional, user-friendly experience. The key principles include:

1. **Balance:** This refers to the distribution of visual weight.
 - **Symmetrical Balance:** Elements are evenly centered.
 - **Asymmetrical Balance:** Uses different scales and colors to create a balanced feel without perfect mirroring.
2. **Contrast:** Using opposing elements—like light vs. dark colors or large vs. small text—to make important elements (like buttons) stand out.
3. **Emphasis (Hierarchy):** Directing the user's eye to the most important part of the page first (e.g., a "Sign Up" button or a headline) using size, color, or placement.
4. **Consistency (Unity):** Keeping fonts, button styles, and color schemes the same across all pages of a website to provide a professional and cohesive feel.

5. **White Space (Negative Space):** The empty space between elements. It prevents the website from looking cluttered and makes content easier to read.
6. **Responsive Design:** Designing graphics that automatically scale and look good on all devices, from large desktop monitors to small smartphone screen

b) What is graphic package? Write down desirable feature of graphics package.

. b) Graphic Package

Definition: A **Graphic Package** is a collection of software tools and libraries that allow a user to create, edit, and manipulate images and models on a computer. It provides the mathematical and visual framework needed to produce computer graphics without the user having to write complex code from scratch.

Desirable Features of a Graphics Package

To be considered effective and professional, a graphics package should have the following features:

- **Ease of Use:** A user-friendly interface (GUI) with logical menus and tools so that both beginners and pros can work efficiently.
- **Support for Multiple Formats:** The ability to import and export files in various formats like **JPEG, PNG, SVG, PDF, and GIF**.
- **Precision and Accuracy:** High-resolution capabilities and the ability to work with exact coordinates (especially important for CAD packages).
- **Transformation Tools:** Built-in functions for **Scaling, Rotating, Translating (moving), and Mirroring** objects.
- **Layering System:** The ability to stack different elements of a design on separate layers so they can be edited independently without affecting the rest of the image.
- **Color Management:** Support for different color models like **RGB** (for screens) and **CMYK** (for printing).
- **Undo/Redo Functionality:** A robust "History" feature that allows users to fix mistakes easily.
- **Library of Shapes/Objects:** A pre-built set of basic shapes (primitives) like lines, circles, and polygons to speed up the design process.

8a. Explain methods of 3D object representation.

a) Methods of 3D Object Representation

3D object representation is how we define and store 3D shapes in computer graphics. Common methods include:

1. **Polygonal Surface Representation**
 - Uses polygons (mostly triangles/quads) to model surfaces.
 - Simple and widely used in games and real-time rendering.
 - Example: Mesh models in Blender or Unity.
2. **Boundary Representation (B-Rep)**
 - Describes objects using their surface boundaries.
 - Defines vertices, edges, and faces.
 - Used in CAD and solid modeling.
 - Example: Polyhedron, ellipsoid, aircraft models.
3. **Space Partitioning (e.g., Octree)**
 - Divides 3D space into smaller cubes (voxels).
 - Efficient for collision detection and volume rendering.
 - Example: Medical imaging, terrain modeling.
4. **Spline and Curve-Based Models**
 - Uses curves (Bezier, B-spline) to define smooth surfaces.
 - Ideal for organic shapes like faces or cars.
5. **Solid Modeling**
 - Represents full volume, not just surface.
 - Includes primitives like cubes, cylinders, spheres.
 - Example: Boolean operations in CAD.

8b.

b) Describe about translation and scaling in 2D transformation with example, deeply

1. 2D Translation (Moving)

Translation is a rigid body transformation that moves objects without deformation. Every point (x, y) is shifted by a distance t_x in x -direction and t_y in y -direction.

Equations: $x' = x + t_x$; $y' = y + t_y$

Matrix (Homogeneous): $P' = T \cdot P \Rightarrow \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$

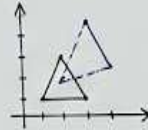
Deep Example: Translate Triangle A(1,1), B(3,1), C(2,3) by vector $T(t_x=2, t_y=3)$.

For A(1,1): $x' = 1 + 2 = 3$; $y' = 1 + 3 = 4$. $\Rightarrow A'(3,4)$

For B(3,1): $x' = 3 + 2 = 5$; $y' = 1 + 3 = 4$. $\Rightarrow B'(5,4)$

For C(2,3): $x' = 2 + 2 = 4$; $y' = 3 + 3 = 6$. $\Rightarrow C'(4,6)$

Final Coordinates: $A'(3,4)$, $B'(5,4)$, $C'(4,6)$



2. 2D Scaling (Resizing)

Scaling changes the size of an object. It expands or compresses dimensions by multiplying coordinates by scaling factors s_x and s_y relative to the origin.

Equations: $x' = x \cdot s_x$; $y' = y \cdot s_y$

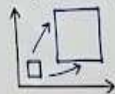
Matrix: $P' = S \cdot P \Rightarrow \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$

Note: If $s_x = s_y$, it's Uniform Scaling (maintains shape). If $s_x \neq s_y$, it's Differential Scaling (distorts shape).

Deep Example: Uniformly scale a square with corner P(2,2) by factor $S(s_x=2, s_y=2)$.

$P(2,2) \rightarrow x' = 2 \cdot 2 = 4$; $y' = 2 \cdot 2 = 4$.

New Point $P'(4,4)$. (Object size doubles, moves further from origin).



9. Write short notes on:

a. **Morphing**

Morphing (short for "metamorphosing") is a special effect in computer animation that changes one image or shape into another through a seamless transition.

- **Process:** It involves two main steps: **warping** (distorting the source image to match the target's shape) and **cross-dissolving** (fading the colors of the first image while bringing in the second).
- **Keyframes:** The user defines "feature points" (like eyes, nose, and mouth) on both images so the computer knows which parts of the first image correspond to the second.
- **Application:** Widely used in movies (e.g., a person turning into a werewolf) and commercial transitions.

b. Parallel Projection

Parallel projection is a method used to represent 3D objects on a 2D screen where the projection lines are parallel to each other.

- **Core Principle:** Unlike perspective projection (where objects get smaller as they move away), parallel projection maintains the relative proportions of objects regardless of their distance from the viewer.
- **Types:**
 1. **Orthographic:** Projection lines are perpendicular to the view plane (used in engineering blueprints).
 2. **Oblique:** Projection lines are not perpendicular to the view plane, showing the front and side views simultaneously.
- **Advantage:** It is extremely useful for technical drawings and CAD because it allows for accurate measurements directly from the drawing.

c. CRT Monitor

The Cathode Ray Tube (CRT) monitor was the standard display technology for computers before the invention of LCD and LED screens.

- **Working Mechanism:** An electron gun at the back of the tube fires a beam of high-speed electrons. These electrons pass through a focusing system and hit a phosphorus-coated screen.
- **Luminescence:** When the electron beam hits the phosphor, it glows, creating a visible dot (pixel) on the screen.
- **Scanning:** The beam moves rapidly from left to right and top to bottom (**Raster Scan**) to refresh the image dozens of times per second.
- **Components:** Key parts include the electron gun, deflection plates (to move the beam), and the shadow mask (for color precision).

REGULAR/BACK EXAM-2077, CHAITRA

1. What is Computer Graphics? Explain Computer Aided Designing (CAD) and Computer Aided Manufacturing (CAM) with example.

Computer Graphics is a branch of computer science that focuses on generating, manipulating, and displaying visual images using computers. It allows users to visualize abstract data or real-world objects in 2D or 3D.

Widely used in animation, gaming, simulations, CAD/CAM, and virtual reality

1. Computer Aided Design (CAD)

Definition.

CAD is the use of **computer software to create, modify, analyze, and optimize designs.**

It replaces manual drafting, allowing **precision, flexibility, and visualization** in designing parts, machines, or structures.

CAD improves **productivity, accuracy, and design quality.**

Objectives of CAD:

- To produce **accurate 2D drawings and 3D models.**
- To **simulate and analyze** product performance before manufacturing.
- To **store, retrieve, and modify designs** easily.

Components of CAD:

- **Hardware:** Computers, graphics terminals, input/output devices.
- **Software:** AutoCAD, SolidWorks, CATIA.
- **Human Resources:** Designers, engineers, operators.

Example:

- Designing a **mechanical component** like a gear in **SolidWorks.**
- Architectural layout of a building using **AutoCAD.**

Advantages:

- Faster and **error-free design.**
- Easy **modifications and simulation.**
- Helps in **documentation and presentation.**

2. Computer Aided Manufacturing (CAM)

Definition

CAM is the use of **computer software and machines to automate the manufacturing process.** It converts **CAD designs into machine instructions** for producing the actual product.

CAM ensures **precision, repeatability, and efficiency** in manufacturing.

Objectives of CAM:

- Reduce **manual labor** in production.
- Increase **accuracy and consistency** of products.
- Integrate with CAD for **seamless design-to-production workflow.**

Components of CAM:

- **Hardware:** CNC machines, robots, 3D printers.
- **Software:** Mastercam, EdgeCAM, Fusion 360.
- **Human Resources:** Machine operators, programmers.

Example:

- CNC milling machine producing a **metal gear** using CAD design input.
- **3D printing a prototype** object from a CAD model.

Advantages:

- Faster production and **high precision.**
- **Reduces material waste.**
- Integrates with CAD for **automated design-to-production process.**

3. Write about the Raster and vector scan display technology.

Raster Scan Display Technology Raster scan display is the most common type used in modern monitors (CRT, LCD, LED, etc.). It works by scanning the screen line by line (from top to bottom) with an electron beam (in CRT) or pixel activation (in flat panels). The screen is divided into a grid of pixels (picture elements), each with RGB color values stored in a frame buffer (video memory).

Key Features of Raster Scan:

- Image is built from a matrix of pixels (bitmap/raster image).
- Refresh rate: Beam scans the entire screen 50–120 times per second to avoid flicker.
- Supports complex images: Photos, shaded areas, colors, textures.
- Uses scan conversion: Algorithms convert lines/polygons into pixel patterns.
- Advantages: High color depth, realistic images, low cost today.
- Disadvantages: Fixed resolution → zooming causes pixelation; aliasing (jagged edges) possible.

Vector Scan Display Technology (also called Random Scan or Calligraphic Display) Vector scan displays (used in early computers like oscilloscopes, CAD systems, e.g., Tektronix terminals) draw images by directing the electron beam directly to the points defining lines/shapes. No fixed grid — beam moves only where lines are needed (from point to point).

Key Features of Vector Scan:

- Image defined by mathematical vectors (lines, curves) → continuous, smooth lines.
- No pixels → no aliasing or pixelation; perfect scaling.
- Refresh: Beam redraws only the lines repeatedly (no full screen scan).
- Advantages: Very sharp lines, high resolution for line drawings, no jagged edges.
- Disadvantages: Cannot fill areas or show complex shaded images; slower for dense drawings; expensive hardware.

4. Explain about the DDA line drawing algorithm.

Q3. Explain about the DDA line drawing algorithm.

DDA is a scan-conversion algorithm for drawing lines that calculates intermediate pixel points by using incremental coordinate changes based on the line's slope. It's simple but slower due to floating-point arithmetic.

Algorithm Steps:

1. Input two endpoints (x_1, y_1) & (x_2, y_2) .
2. Calculate differences: $\Delta x = x_2 - x_1$, $\Delta y = y_2 - y_1$.
3. Determine number of steps: $\text{Steps} = \max(|\Delta x|, |\Delta y|)$.
4. Calculate increments: $x_{\text{inc}} = \Delta x / \text{Steps}$, $y_{\text{inc}} = \Delta y / \text{Steps}$.
5. Initialize starting point: $x = x_1$, $y = y_1$.
Plot Pixel(Round(x), Round(y)).
6. Iterate 'Steps' times: $x = x + x_{\text{inc}}$, $y = y + y_{\text{inc}}$.
Plot Pixel(Round(x), Round(y)).

Formula: $\Delta x = x_2 - x_1$, $\Delta y = y_2 - y_1$
 $\text{Steps} = \max(|\Delta x|, |\Delta y|)$
 $x_{\text{inc}} = \Delta x / \text{Steps}$, $y_{\text{inc}} = \Delta y / \text{Steps}$
 $x_{k+1} = x_k + x_{\text{inc}}$
 $y_{k+1} = y_k + y_{\text{inc}}$

Example: Draw line from (2,3) to (8,6)

$x_1 = 2$, $y_1 = 3$; $x_2 = 8$, $y_2 = 6$
 $\Delta x = 8 - 2 = 6$, $\Delta y = 6 - 3 = 3$
 $\text{Steps} = \max(|6|, |3|) = 6$
 $x_{\text{inc}} = 6/6 = 1.0$, $y_{\text{inc}} = 3/6 = 0.5$

Step-by-Step Points:

Start: (2, 3). Plot (2, 3).
 $k=1$: $x = 2 + 1.0 = 3.0$, $y = 3 + 0.5 = 3.5$. Plot (3, 4).
 $k=2$: $x = 3.0 + 1.0 = 4.0$, $y = 3.5 + 0.5 = 4.0$. Plot (4, 4).
 $k=3$: $x = 4.0 + 1.0 = 5.0$, $y = 4.0 + 0.5 = 4.5$. Plot (5, 5).
 $k=4$: $x = 5.0 + 1.0 = 6.0$, $y = 4.5 + 0.5 = 5.0$. Plot (6, 5).
 $k=5$: $x = 6.0 + 1.0 = 7.0$, $y = 5.0 + 0.5 = 5.5$. Plot (7, 6).
 $k=6$: $x = 7.0 + 1.0 = 8.0$, $y = 5.5 + 0.5 = 6.0$. Plot (8, 6).

Result/Table:

Step (k)	Calculated (x, y)	Plot Pixel (Round(x), Round(y))
0	(2, 3)	(2, 3)
1	(3.0, 3.5)	(3, 4)
2	(4.0, 4.0)	(4, 4)
3	(5.0, 4.5)	(5, 5)
4	(6.0, 5.0)	(6, 5)
5	(7.0, 5.5)	(7, 6)
6	(8.0, 6.0)	(8, 6)

Note: DDA uses floating-point arithmetic which is slower and less accurate than integer-based Bresenham's algorithm.

4. Derive ellipse drawing algorithm.

Date: 2080/03/05
Page: _____

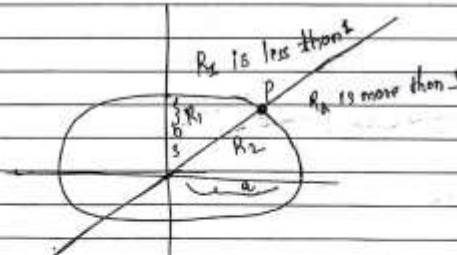
Mid Point Ellipse Drawing Algorithm

Eqⁿ of ellipse -

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

$$\text{or, } x^2 b^2 + y^2 a^2 = a^2 b^2$$

$$\text{or, } x^2 b^2 + y^2 a^2 - a^2 b^2 = 0$$

$$f_e(x, y) = x^2 b^2 + y^2 a^2 - a^2 b^2 \quad \text{--- eqn (1)}$$


At point 'P' - eqn (1) derivative is

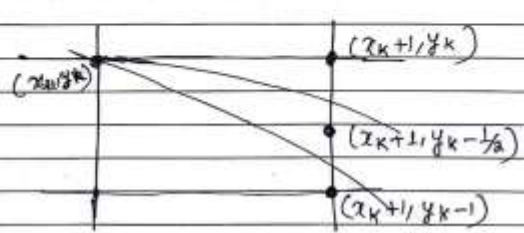
$$2xb^2 + 2ya^2 \frac{dy}{dx} = 0$$

$$2xb^2 + 2ya^2 = 0 \quad \left[\text{for this point 'P'} \right]$$

$$\text{or, } 2xb^2 \leq 2ya^2 \quad \left[\frac{dy}{dx} \text{ is } -1 \right]$$

Date: _____
Page: _____

Region 1 -

$$2ab^2 \leq 2ya^2$$


For Region 1 (R_1)

Let (x_k, y_k) be the current pixel, then the candidate pixels are

$$(x_k+1, y_k) \text{ and } (x_k+1, y_k-1)$$

Mid-point = $(x_k+1, y_k-1/2)$

We know, the function of ellipse at the mid-point is the decision parameter.

5. Define projection. Explain the types of projection with examples.

• Projection is a method of representing a 3D object on a 2D plane (like paper or screen). It is widely used in engineering drawing, computer graphics, and design to visualize objects.

Types of Projection

1 Orthographic Projection

- Definition: A method where the object is projected onto a plane using parallel lines perpendicular to the plane.
- Characteristics:
- Shows true dimensions.
- Multiple views (front, top, side) are drawn.
- Example: Engineering drawings of machines use orthographic projection.

2 Oblique Projection

- Definition: The object is projected using parallel lines that are not perpendicular to the plane.
- Characteristics:
- Front view is drawn in true shape.
- Depth is represented at an angle (commonly 45°).
- Example: Cabinet and cavalier drawings in design.

3 Perspective Projection

- **Definition: Projection** where lines converge at a point (called the vanishing point), giving a realistic view.
- Characteristics:
- Mimics human eye vision.
- Objects farther away appear smaller.
- Example: Architectural drawings, 3D art, road drawings.

4 Isometric Projection.

- Definition: A type of axonometric projection where the object is rotated so that all three axes are equally inclined (120° apart).
- Characteristics:
- Dimensions along all axes are scaled equally.
- Gives a 3D view without distortion.
- Example: Isometric drawings of mechanical parts.

6. Explain how 3-D objects are presented Discuss about polygon table and polygon Surface.

D Object Representation (Short 3-line note)

3D objects in computer graphics are represented using mathematical models that define their shape, position and surface in three-dimensional space (x, y, z). The most common and widely used method is **polygonal representation** (boundary representation or B-rep), where curved/complex surfaces are approximated by a large number of flat polygons (mostly triangles or quadrilaterals). This approach is efficient for rendering, lighting, transformations, clipping and hidden surface removal in modern graphics pipelines.

Polygon Surface

- A 3D object's surface is modeled as a collection of connected polygons forming a closed mesh.
- Polygons are usually triangles (most GPU-friendly) or quadrilaterals.
- Each polygon has: vertices, edges, normal vector (for lighting), material properties (color, texture).
- Advantages: Simple math, fast hardware rendering, supports shading (Gouraud, Phong).
- Disadvantage: Curved objects need many small polygons for smoothness (high poly count)

Polygon Tables

To store the geometry of these 3D objects efficiently, we use **Polygon Tables**. These tables organize the data into three specific lists:

1. **Vertex Table:** Stores the (x, y, z) coordinates for every unique point (vertex) in the object.
2. **Edge Table:** Stores pointers to the Vertex Table to show which two points form a line (edge).
3. **Polygon-Surface (Face) Table:** Stores pointers to the Edge Table to show which edges form a complete flat surface (face).

Why use tables? It prevents redundancy. If a vertex is shared by four different polygons, its coordinates are only stored once in the Vertex Table, saving memory and making updates easier

7. Explain the Surface detection technique in brief.

Surface Detection Techniques

Surface Detection (also called **Visible Surface Detection** or **Hidden Surface Removal**) is the process used to determine which parts of a 3D scene are visible from a particular viewpoint and which parts are hidden behind other objects.

There are two main categories:

1. Object-Space Methods

These compare objects and their parts with each other to decide visibility.

- **Back-Face Culling:** This is the simplest method. It calculates the "normal vector" (direction) of each polygon. If the polygon is facing away from the camera, it is not drawn. This usually removes about 50% of the polygons in a scene instantly.

2. Image-Space Methods

These decide visibility on a pixel-by-pixel basis.

- **Z-Buffer Algorithm (Depth-Buffer):** This is the most widely used technique. For every pixel on the screen, the computer stores a "depth value" (Z-value) in a buffer. When a new polygon is rendered, its depth is compared to the stored value. If the new polygon is closer to the camera, it replaces the old one; otherwise, it is discarded.
- **Scan-Line Method:** This algorithm processes the scene line-by-line (from top to bottom). It determines which surfaces are visible for each horizontal line on the screen.

8. What is Animation sequence? Explain Morphing technique.

Animation Sequence

An animation sequence refers to the structured, step-by-step process of creating a digital motion picture. Instead of just drawing random frames, animators follow a logical pipeline to ensure the motion is smooth and the story is clear.

Morphing Technique

Morphing (short for "Metamorphosing") is a computer graphics technique used to transform one image or shape into another through a seamless, fluid transition. It is widely used in movies and advertisements.

How Morphing Works:

Morphing is achieved by combining two distinct processes:

1. **Warping (Distortion):** * The animator selects **Feature Points** (or Control Points) on both the source image (e.g., a man's face) and the target image (e.g., a lion's face).
 - The software distorts the shapes so that the eyes, nose, and mouth of the man align with the eyes, nose, and mouth of the lion.
2. **Cross-Dissolving (Fading):**
 - While the shapes are warping, the first image gradually fades out (becomes transparent) while the second image gradually fades in.

Key Features:

- **Feature Points:** These are crucial for a realistic morph. If points are mismatched, the transition looks "broken."
- **Applications:** Transforming a character into a monster, aging a person's face, or changing one car model into another in a commercial.

