Subject:
Date:

6th batch final

1. In circle drawing, discuss the uncertain case that Lead to the development of "Midpoint circle Algorithm"

Uncertain case:

Answer: At each step, there are two possible pixel consider the current point (x, y)

1. $(x+1, y)$
2. $(x+1, y-1)$

The uncertain case is deciding which pixel to choose, because the actual circle lies between these two pixels. It is not obvious which one is closer to the real curve.

This ambiguity is what we Call ~~called~~ the uncertain case.

Solve: To solve this, the mid circle algorithm was devloped. It use decision parameter to check which pixel is closer to the circle by evaluating the midpoint between the two choice. If the midpoint is inside the circle, choose

1. $(x+1, y)$

Otherwise,

2. $(x+1, y-1)$

This method avoid floating point and calculation using only integer arithmatic. making it fast and effic...

Method: Evalutes $f(x,y) = x^2 + y^2 - r^2$ at the midpoint to decide the next point, ensuring efficiency.

$$f(x,y) \begin{cases} <0 \rightarrow \text{inside the circle} \\ =0 \rightarrow \text{on the circle path} \\ >0 \rightarrow \text{outside the circle} \end{cases}$$

intial position

$$P_k = 1 - r = 1 - 10 = -9$$

$$\boxed{\begin{matrix} x_0 = 0 & \& & y_0 = r \\ x_0 = 0 & \& & y_0 = 10 \end{matrix}}$$

(compare P_k value) condition:

i) if, $p<0$ then $P_{k+1} = P_k + 2x_{k+1} + 1$ $(x+1, y)$

(ii) if, $p \geq 0$ then $P_{k+1} = P_k + 2x_{k+1} + 1 - 2y_{k+1}$ $(x+1, y-1)$

k	P_k	x_{k+1}, y_{k+1}	$2x_{k+1}$	$2y_{k+1}$
0	-9	1, 10	2	20
1	-6	2, 10	4	20
2	-1	3, 10	6	20

p initial value 0

[illegible] value same [illegible]

Subj:

Date :

(b) consider endpoints $P_1(0,0)$ and $P_2(4,6)$. Based on Digital differential Analyzer (DDA), examine the points that make up the line P_1P_2.

before (2)

chosen-sutherland

Top→TBRL—Left
Bottom
Right

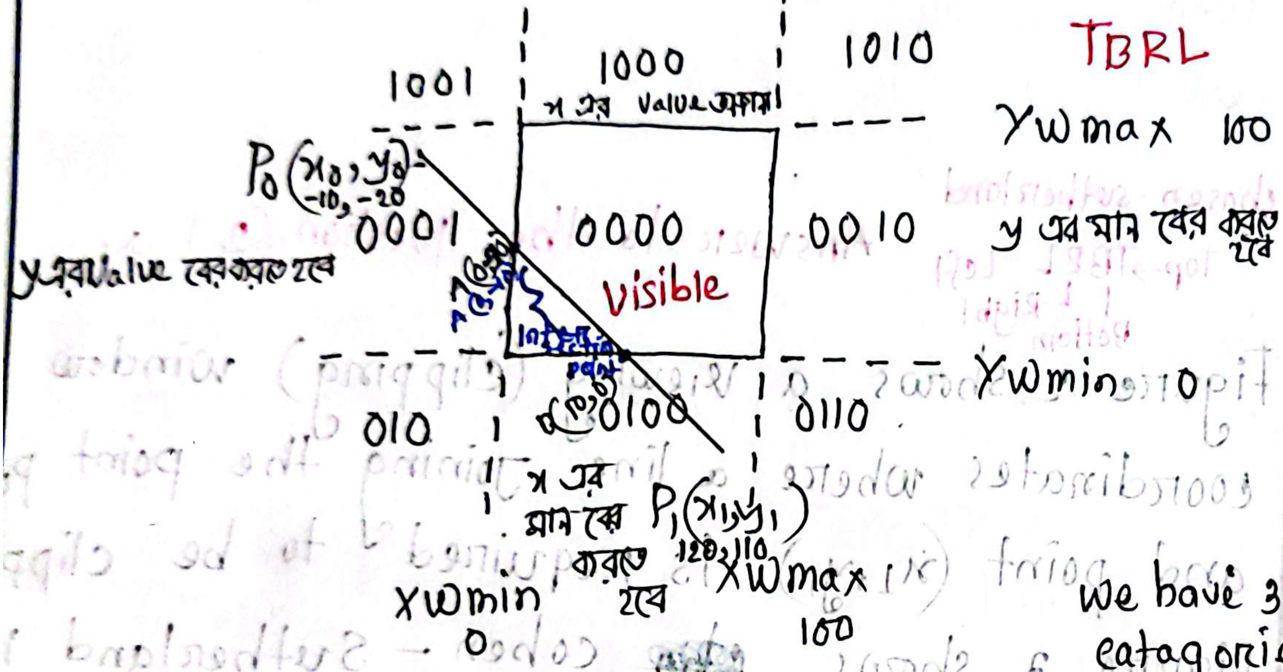
Answer to the question (2) a.

Figure 2 shows a viewing (clipping) window with coordinates where a line joining the point $P_0(x_0, y_0)$ and point (x_1, y_1) is required to be clipped. Figure 3 shows cohen-Sutherland line clipping codes and algorithm to find the intersection points that form the line after clipping.

Subject:
Date:

(a) Demonstrate the process to find the intersection points that form the line after the clipping using the cohen-sutherland algorithm with
$xw_{min} = 0$, $xw_{max} = 100$, $yw_{min} = 0$, $yw_{max} = 100$

$x_0 = -10$, $y_0 = -20$, $x_1 = 120$, $y_1 = 110$



We have 3 clipping categories
1. Visible
2. Not Visible
3. Clipping candidate

Here, given ~~below~~

$x_0 = -10$
$y_0 = -20$
$x_1 = 120$
$y_1 = 110$

Let a line with endpoint A (120, 110), B (-10, -20)

End point P_0	region	0001
P_1	"	0100
Bitwise AND	=	0000

So it is clipping candidate

Date :

finding intersection point

(1) y Value of vertical line :

$$m = \frac{y - y_1}{x - x_1}$$

$$\Rightarrow y - y_1 = m(x - x_1)$$

$$\text{বা}, \; y = y_1 + m(x - x_1)$$

$x = x\,max$ or $x\,min$

$y = y\,max$ or $y\,min$

(II) x Value of Horizontal Line

$$m = \frac{y - y_1}{x - x_1}$$

$$\text{বা}, \; m(x - x_1) = y - y_1$$

$$\text{বা}, \; x - x_1 = \frac{y - y_1}{m}$$

$$\text{বা}, \; x = x_1 + \frac{y - y_1}{m}$$

~~FOR P_1 =~~

FOR P_0 : $x = 0$

$$y = y_0 + m(x - x_0)$$

$$y = y_1 + m(x - x_1)$$

$$= -20 + 1(0 + 10)$$

$$= -20 + 10$$

$$= -10$$

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

$$= \frac{110 + 20}{120 + 10}$$

$$= \frac{130}{130}$$

$$= 1$$

FOR P_1 : $y = 0$

$$x = x_0 + \frac{y - y_0}{m}$$

$$= x_1 + \frac{y - y_1}{m}$$

$$= 120 + \frac{0 - 110}{1}$$

$$= 120 + \left(\frac{-110}{1}\right)$$

$$= 120 - 110$$

$$= 10$$

Subject:
Date:

(b) How viewing-coordinate reference frame can be es[tablished]

A viewing-coordinate reference frame is estabished by defining an (i) orgin,

(ii) Select view window (xwmin, xwmax, y[wmin], ywmax)

(iii) Translate and rotate the world so th[at] the window is aligned with viewing coordinate axes.

Answer to the Question: 3 (a)

Distinguish the terms (i)

Brightness	Luminance
1. Subjective → depends on how bright it looks to the eye.	1. Objective - measured light intensity from source of surface
2. Affected by perception, background, and eye adaption	2. Independent of perception, purely a physical quantity
3. Used in human-centered application	3. Used in scien[tific] technical fields (physics)
4. Not directly measurment with instruments, depends on individual perception	4. Measureable with a photometer, typica[lly] in candles per square meter (cd/m²)

Date:

Hue

1. The attribute of color that distinguishes it as red, blue, green

II. Represents the type of colour determined by the dominant wavelength light.

III. It describes how the color looks

IV. used in color theory, design, art

~~Dominal~~ Dominant

~~1: Represents the type of color, determined by dominant wavelength of light.~~

1. The single wavelength of light that most closely match the perceived hue.

2. used in color science to specify the hue based on spectral properties

III. It describes how the color is measured

IV. used in optics and color mesurements properties

Saturation

1. The intensity or vividness of a color, indicating how pure or mixed it is with white/gray (Describes how vivid or dull a color appears)

2. High Saturation means vivid, pure color
Low " " grayish

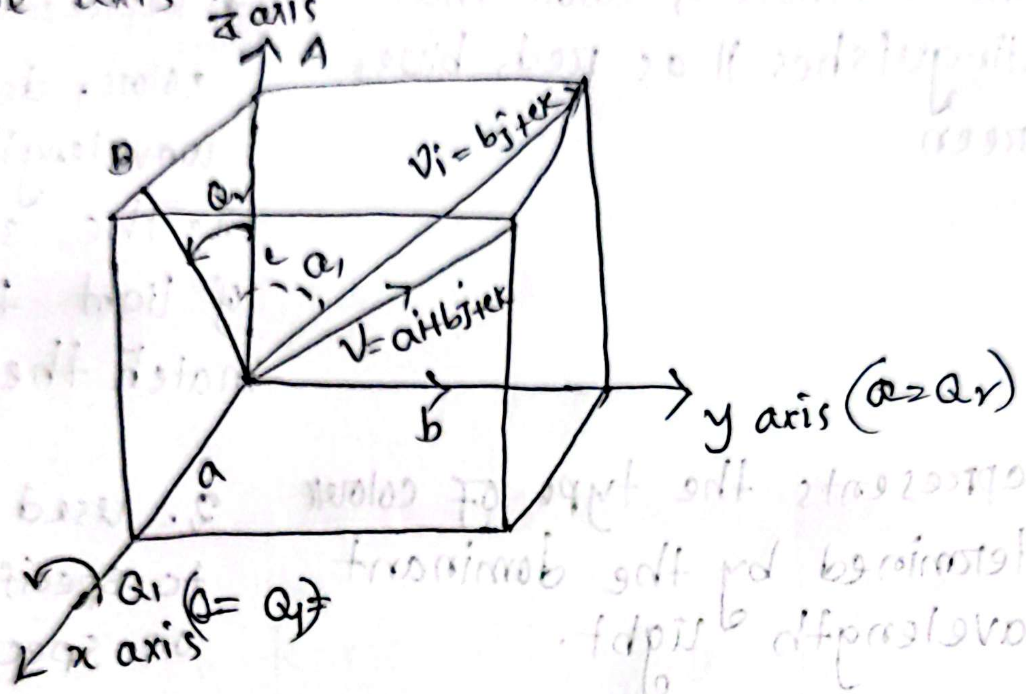
Exciatation

1. measure how much pure color is present in white-light mixture

II. quantifies how close a color is to a pure spectral color versus white light.

Subject:
Date:

(b) Find a transformation A_v which alligns a given vector V with the vector k along the positive axis:



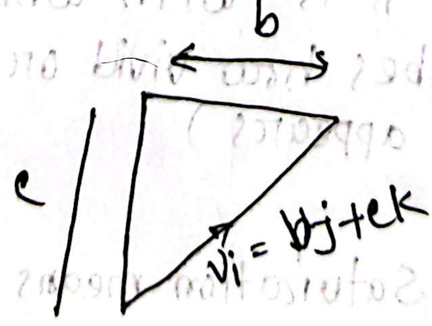
given vector is $V = ai+bj+ck$

$a=0 \quad b=1 \quad c=1$

$|V| = \sqrt{a^2+b^2+c^2}$

Step 1: Rotation of vector V about x axis

Now,
$V_1 = bj+ck$
$|V_1| = \sqrt{b^2+c^2}$



from figure, $\sin\theta_1 = b/\sqrt{b^2+c^2}$

$\cos\theta_1 = c/\sqrt{b^2+c^2}$

Subject:
Date:

Let, $\sqrt{b^2+c^2} = d$

$\sin\theta_1 = \frac{b}{d}$

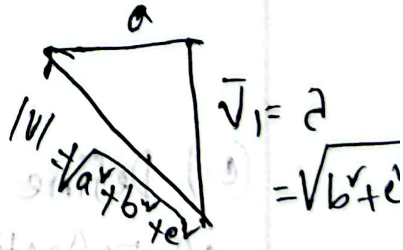
$\cos\theta_1 = c/d$

Now the rotation matrix about x axis

$$R(x) = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$Rx = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{c}{d} & \frac{b}{d} & 0 \\ 0 & -\frac{b}{d} & \frac{c}{d} & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Rotate Vector V_1 about y axis



$\sin\theta_2 = a/\sqrt{a^2+b^2+c^2} = a/|V|$

$\cos\theta_2 = \sqrt{b^2+c^2}/\sqrt{a^2+b^2+c^2} = d/|V|$

Now Rotational matrix about y axis

$$R(y) = \begin{vmatrix} \cos\theta & 0 & -\sin(-\theta) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(-\theta) & 0 & \cos(-\theta) & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Subject:..........................

Date:..........................

$$Ry = \begin{vmatrix} \frac{d}{|V|} & 0 & \frac{a}{|V|} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{-a}{|V|} & 0 & \frac{d}{|V|} & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Now combined transform

$$|A\,V| = [Rx]\,[Ry]$$

$$= \begin{vmatrix} \frac{d}{|V|} & 0 & \frac{a}{|V|} & 0 \\ \frac{-ab}{d|V|} & \frac{c}{d} & \frac{b}{|V|} & 0 \\ \frac{-ac}{d|V|} & \frac{-b}{d} & \frac{c}{|V|} & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

This is the matrix which aligns a given Vector $V = ai + bj + ck$ with the Vector K along positive z axis.

(c) Define 2D mirror reflection. write the matrix form of reflection when an object is reflected with respect to x axis

In computer graphics, 2D reflection is a technique that involves mirroring ~~&~~ or flipping an object or coordinate system across a specific axis in a 2D plane.

1. Reflection about the x axis: The object can be reflected on the x-axis with the help of the following matrix $\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

Subject:
Date:

Answer to the question 4.(a)

write down the difference between geometric transformation and coordinate transformation

Geometric	coordinate
I. change the shape, size or position of object	I. Manipulates the coordinates of object (স্থানাঙ্ক)
II. collects object properties	II. Does not preserve object properties
III. used in computer graphics, animation	III. Applied in navigation system, robotics.

(b) For scalling an object, is it necessary to have fixed point?

No, it is not always necessary to have a fixed point when scalling an object. ~~While~~

Scalling is usually performed with respect to a fixed point, commonly the orgin (0,0). If scalling is done without considering a fixed point, the object may also shift position, not just change size.

Subject:..................
Date:..................

(c) Explain how to convert standard 3D coordinates (x, y, z), to homogeneous coordinates, and how to convert homogenous coordinates to Standard 3D coordinates.

convert standard 3D coordinates (x, y, z) to homogenous coordinates

1. Add fourth coordinate

Standard 3D point (x, y, z) and create the homoge

Add fourth coordinate w coordinate (w, x, y, z)

2. Set w to a value: A common choice for w is

1. Let, $w = 1$

so, homogenous coordinates $(x, y, z, 1)$

$\therefore$ The homogenous coordinates representing the point (x, y, z) are $(x, y, z, 1)$

Subject:
Date :

Homogeneous coordinates to standard 3D coordinates

Divided by w : Homogeneous coordinates (x, y, z, w)

$$= \frac{x, y, z, w}{w} = \frac{x}{w}, \frac{y}{w}, \frac{z}{w}, \frac{w}{w}$$

$$= x, y, z \text{ (3D coordinates)}$$

(d) Define 2D Scalling and write matrix form scalling when an object is 1/2 with respect to x - axis.

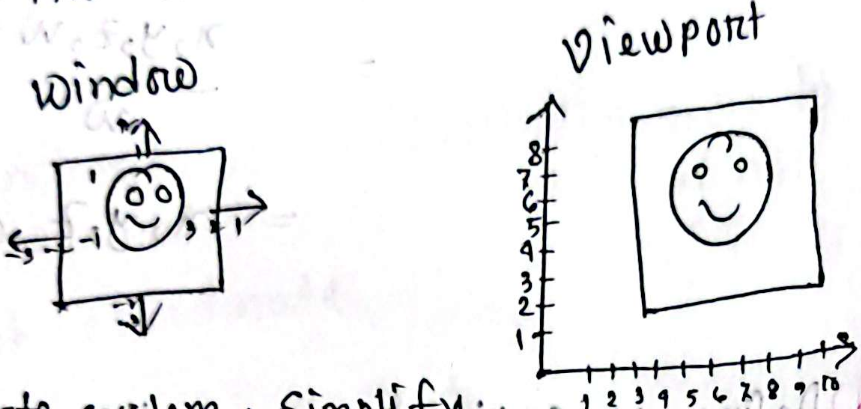
2D scalling: 2D Scalling is a geometry transformation that change the size of an object in the 2D plan. It modifies the object's dimenson along the x and y axes using scalling factor. If the scalling factor is grater than 1, the object enlarge, If it is between 0 and 1 the object is shrink

Matrix from scalling when object 1/2 with respect x axsis

$$\begin{vmatrix} s_x & 0 \\ 0 & s_y \end{vmatrix} = \begin{vmatrix} 1/2 & 0 \\ 0 & 1 \end{vmatrix} \Rightarrow \begin{vmatrix} S_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 1/2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{vmatrix}$$

Answer to the question 6: (a)

Figure 1 shows an object in a world coordinate system and its transformation in a screen system



screen coordinate system. Simplify your Vector obtained from the word to screen transform Vector with the information

(a) Derive a world-to-screen transform Vector based on a generic coordinate (x, y)

Derivation a world-to-screen transform Vector

Let, world coordinates (x, y)

where $x = x\ max$ or $x\ min$

$y = y\ max$ or $y\ min$

Screen coordinates (x_s, y_s)

where scree resolution

w = width, Height = h

Subject:.................................
Date:.................................

Normalize world coordinates

$$x_n = \frac{x - x_{min}}{x_{max} - x_{min}}$$

$$y_n = \frac{y - y_{min}}{y_{max} - y_{min}}$$

convert screen coordinates

$$x_s = x_n \times w$$
$$= \frac{x - x_{min}}{x_{max} - x_{min}} \times w$$

$$y_s = (1 - y_n) \times h$$
$$= \left(1 - \frac{y - y_{min}}{y_{max} - y_{min}}\right) \times h$$

Final screen vector

Screen $\begin{bmatrix} x_s \\ y_s \end{bmatrix}$

OR,

Subject:
Date:

World coordinate (x, y)

Screen coordinate (x_s, y_s) | screen has width w, height h

Camera : centered at (c_x, c_y) in world coordinates with scale factor s

Goal : Transform (x, y) to screen point (x_s, y_s)

Derivation:

Translate to camera : → relative to camera

$$x_{rel} = x - c_x$$
$$y_{rel} = y - c_y$$

Scale to pixel :

$$x_{pixel} = x_{rel} \cdot s$$
$$= s(x - c_x)$$
$$y_{pixel} = s(y - c_y)$$

Shift to screen center : ~~Move~~

$$x_s = x_{pixel} + \frac{w}{2}$$
$$= s(x - c_x) + \frac{w}{2}$$

$$y_s = -y_{pixel} + \frac{H}{2}$$
$$= -s(y - c_y) + \frac{H}{2}$$

Date :

Final transform Vector

$$P_s = \begin{bmatrix} x_s \\ y_s \end{bmatrix} = \begin{pmatrix} S(x-c_x) + \frac{W}{2} \\ -S(y-c_y) + \frac{H}{2} \end{pmatrix}$$

Ans

(b) Suppose that $X_{min} = -3$, $Y_{min} = -3$, $X_{max} = 2$, $Y_{max} = 1$ for the world coordinate system and $U_{min} = 30$, $V_{min} = 10$, $U_{max} = 80$ and $V_{max} = 30$ for the screen coordinate system, simplify your vector obtained from the world-to-screen transform vector with the information.

window : $(X_{min}, Y_{min}) = (-3, -3)$
$(x_{max}, Y_{max}) = (2, 1)$

Viewport $(U_{min}, V_{min}) = (30, 10)$
$U_{max}, V_{max} = (80, 30)$

plugging the values into the equation

$$P_s = \begin{bmatrix} (x - X_{min}) \cdot \frac{U_{max} - U_{min}}{X_{max} - X_{min}} + U_{min} \\ (y - Y_{min}) \cdot \frac{V_{max} - V_{min}}{Y_{max} - Y_{min}} + V_{min} \end{bmatrix}$$

$$P_s = \begin{bmatrix} x-(-3) \times \frac{80-30}{2-(-3)} + 30 \\ y-(-3) \times \frac{30-10}{1-(-3)} + 10 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} (x+3) \times \frac{50}{5} + 30 \\ (y+3) \times \frac{20}{4} + 10 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} (x+3) \times 10 + 3 \\ (y+3) \times 5 + 10 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} (10x+30) + 30 \\ (5y+15) + 10 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 10x+60 \\ 5y+25 \\ 1 \end{bmatrix} \quad \text{———} (1)$$

putting values in this equation

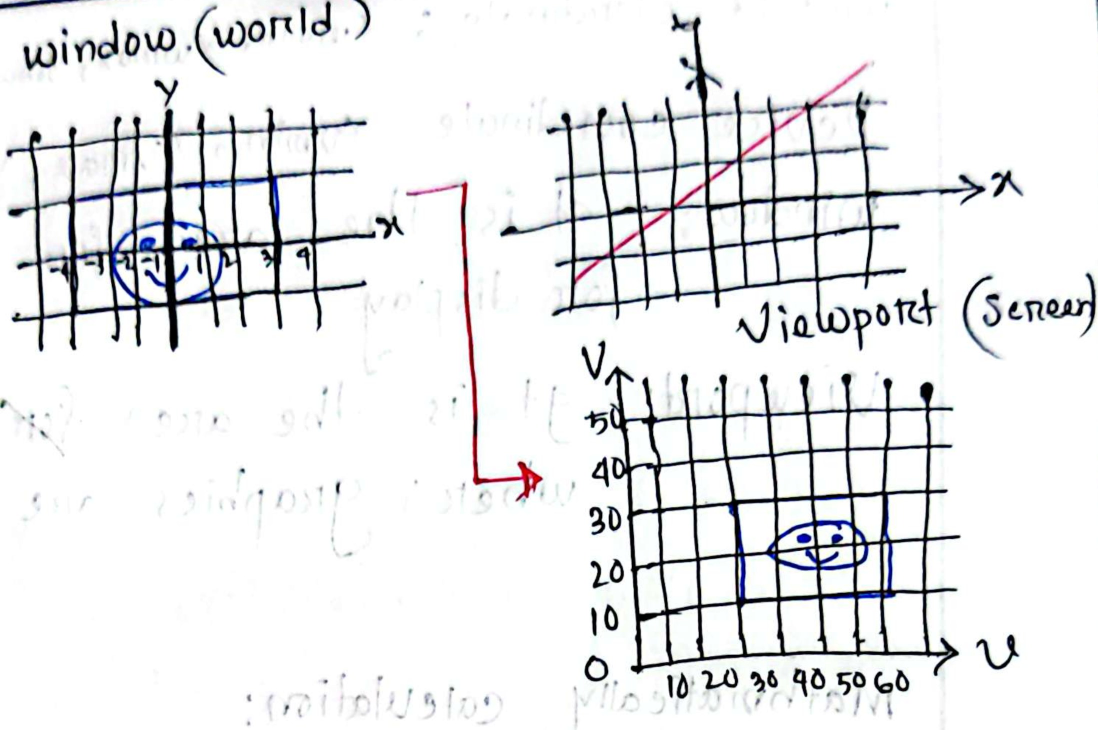
$(x_{min}, y_{min}) = (-3,-3) = (30, 10)$

$(x_{max}, y_{max}) = (2,1) = (80, 30)$

Left eye $(-1,-8) = (50,21)$

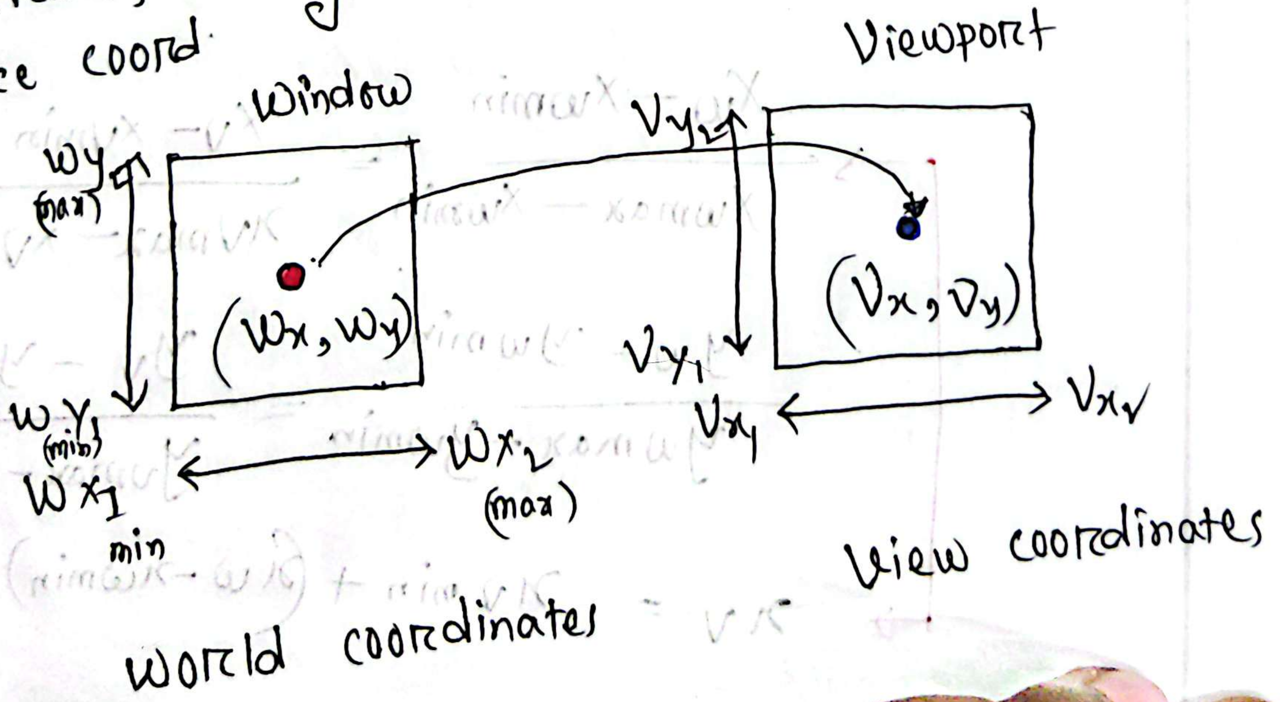
Top of head $\Rightarrow (-0.5, 0.5) = (55, 27.5)$

Subject:
Date:



(c) Analyze the term "Viewing transformation" using a suitable figure.

window to viewport transformation is the process of transforming 2D world-coordinate objects to device coord.



Subject:
Date :

World coordinate : $x_{wmin}, x_{wmax}, y_{max}, y_{wmin}$

Device coordinate $x_{vmin}, x_{vmax}, y_{vmax}, y_{vmin}$

Window: - It is the area for world coordinate se...
for display

- Viewport: It is the area for on the device co...
when graphics are to be displayed

Mathmatically calculation:

Normalized point on window $\left(\frac{x_w - x_{wmin}}{x_{wmax} - x_{wmin}}, \frac{y_w - y_{wmin}}{y_{wmax} - y_{wmin}} \right)$

normalized point on viewport $\left(\frac{x_v - x_{vmin}}{x_{vmax} - x_{vmin}}, \frac{y_v - y_{vmin}}{y_{vmax} - y_{vmin}} \right)$

$$\frac{x_w - x_{wmin}}{x_{wmax} - x_{wmin}} = \frac{x_v - x_{vmin}}{x_{vmax} - x_{vmin}}$$

$$\frac{y_w - y_{wmin}}{y_{wmax} - y_{wmin}} = \frac{y_v - y_{vmin}}{y_{vmax} - y_{vmin}}$$

$$x_v = x_{vmin} + (x_w - x_{wmin}) S_x$$

Subject: ....
Date: ....

$$\frac{y_w - y_{wmin}}{y_{wmax} - y_{wmin}} = \frac{y_v - y_{vmin}}{y_{vmax} - y_{vmin}}$$

বা, $(y_v - y_{vmin})(y_{wmax} - y_{wmin}) = (y_w - y_{wmin}) \times (y_{vmax} - y_{vmin})$

বা, $y_v - y_{vmin} = \frac{(y_w - y_{wmin})(y_{vmax} - y_{vmin})}{y_{wmax} - y_{wmin}}$

বা, $y_v = y_{vmin} + (y_w - y_{wmin}) s_y$

$$s_x = \frac{x_{vmax} - x_{vmin}}{x_{wmax} - x_{wmin}} \qquad s_y = \frac{y_{vmax} - y_{vmin}}{y_{wmax} - y_{wmin}}$$

Answer to the question 6.(a)

Analyze the cause of peak response for RGB colors in our human visual system.

our human visual system is most sensitive to three specific colors : Red, green, blue . This sensitivety is because our eyes have special cells, called cones that are tuned to respond to different wavelength of light.

Subject:

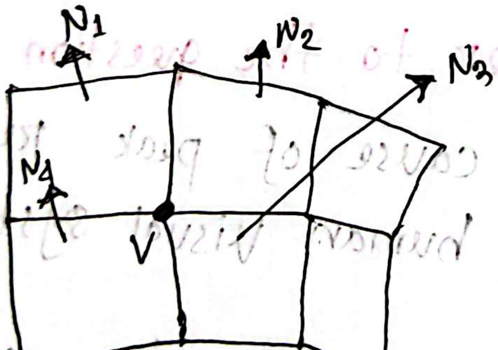
Date :

The peak response to red, green and blue colors occurs because the wavelengths correspond to the optimal stimulation of the three types of cone cells in our eyes.

(b) Differentiate the terms with suitable figure (i) Fore shortening (ii) Vanishing points (iii) View confusion, and (iv) Topological distortion

before

(c) Demonstrate the gouraud Surface Rendering Algorithm with suitable figure (s).



~~Gouraud~~
Gouraud shading is a surface rendering algorithm in computer graphics that produces continous shading by interpolating intensity values across the surface of polygon.

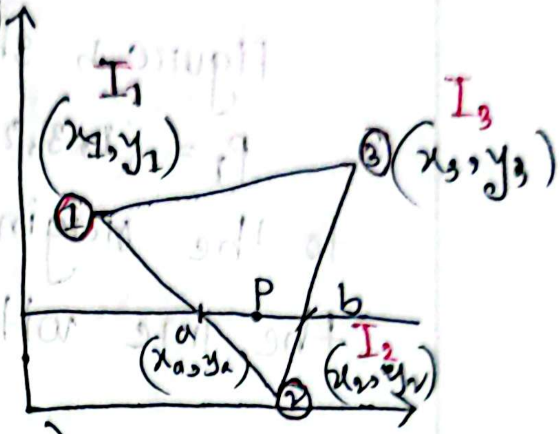
অনুমান করে হিসাব করা (mid value)

Subject:

Date :

Gouraud Shading algorithm

(i) Determine average unit normal vector at polygon vertex



$$N_V = \frac{N_1 + N_2 + N_3}{|N_1 + N_2 + N_3|}$$

$$= \frac{\sum_{i=1}^{n} N}{\left|\sum_{i=1}^{n} N\right|}$$ ($n \rightarrow$ number of surface of polygon)

(ii) By illumination we get intensity of each vertex

$\rightarrow$ Intensity of a : $I_a = \frac{y_a - y_2}{y_1 - y_2} I_1 + \frac{y_1 - y_a}{y_1 - y_2} I_2$

Intensity of b : $I_b = \frac{y_b - y_2}{y_2 - y_3} + \frac{y_3 - y_b}{y_1 - y_2} I_2$

(iii) Intensity at point p is given by :

$$I_p = \frac{x_b - x_p}{x_b - x_a} I_a + \frac{x_p - x_a}{x_b - x_a} I_b$$

Subject:
Date:

Answer to the question 7(a)

Figure 5 shows a line passing through the points $P_1 = (1, 3, 2)$ and $P_2 = (2, 4, 3)$. Translate the line to the origin by setting P_1 to $(0, 0, 0)$. Then rotate the line with about $\theta = 45°$ about y axis

Translate the line to the origin

$P_1 = (1, 3, 2)$
$P_2 = (2, 4, 3)$
$P_1' = (0, 0, 0)$

$P_2' = P_2 - P_1$
$= (2-1,\ 4-3,\ 3-2)$
$= (1, 1, 1)$

Rotate the line about y axis, $\theta = 45°$

$$R(y)\theta = \begin{bmatrix} \cos\frac{\pi}{4} & 0 & \sin\frac{\pi}{4} \\ 0 & 1 & 0 \\ -\sin\frac{\pi}{4} & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} \\ 0 & 1 & 0 \\ -\frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} \end{bmatrix}$$

Date:

Apply the rotation to P_2'

$$P_2'' = R_y(\theta) \cdot P_2'$$
$$= R_y(45^\circ) \cdot P_2'$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} \\ 0 & 1 & 0 \\ -\frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2} \\ 1 \\ -\frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2} \end{bmatrix} = \begin{bmatrix} \sqrt{2} \\ 1 \\ 0 \end{bmatrix}$$

So the line passing through the points $P_1 = (1, 3, 2)$ and $P_2 = (2, 4, 3)$, when translated to origin and about the y axis by $\theta = 45^\circ$, result in a line passing through the origin and the point $(\sqrt{2}, 1, 0)$

Subject:
Date:

(b) Why lighting is important for computer graphic[s]

Lighting is important in computer graphics it adds dept[h], dimension, and mood to scense by showing how obje[cts] interact with light, ~~creating~~ shadows, highlights. It help[s] viewers ~~to~~ understand the shape, position and text[ure] of objects.

(c) convert the RGB color model to YIQ color model

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.144 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\therefore Y = 0.299R + 0.587G + 0.114B$$

$$I = 0.596R - 0.274G - 0.322B$$

$$Q = 0.211R - 0.523G + 0.312B$$

Subject:..............................

Date:..............................

Answer to the question 8.(a)

Define the difference between Gouraud and phong Surface Renedering.

Gouraud	Phong surface
1. Light is calculated at Vertices only	1. Light is calculated at each pixel
2. Performanc faster	2. performance slower
3. Interpolates color across the Surface	3. Interpolates normal Vector
4. Good for basic shading, Low cost system	4. used in modern 3D renedering for relism
5. Visual quality Low	5. Visual quality high
6. It gives less accurate result	6. It gives more accurate result.
7. Requires average processing and less time	7. Require high processing and more time
8. Vertex based interpolation	8. pixel based interpolation

Subject: ....

Date: ....

(b) A perspective transformation is determined by prescribing a center of projection and a view plane. In Figure 6, the object p is Located in world coordinates at (x, y, z). The problem is to determine the image point coordinates at (x, y, z). The problem is to determine the image point coordinates P′(x′, y′, z), as described in Figure 6.

$$\frac{AB}{BC} = \frac{A'B'}{B'C}$$

$$\frac{x}{z} = \frac{x'}{-d}$$

$$\text{বা } x' = -\frac{dx}{z} = \frac{x}{-\left(\frac{z}{d}\right)}$$

$$y' = \frac{y}{-\left(\frac{z}{d}\right)}$$

$$z' = -d$$

$$(x', y', z', 1) \Rightarrow \left(\frac{x}{-(z/d)}, \frac{y}{-(z/d)}, -d, 1\right)$$

Projective transformation

$$\begin{pmatrix} & \end{pmatrix}\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{x}{-\left(\frac{z}{d}\right)} \\ \frac{y}{-\frac{z}{d}} \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} & \end{pmatrix}\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \\ -\frac{z}{d} \end{pmatrix}$$

$$\begin{pmatrix} x \\ y \\ z \\ -\frac{z}{d} \end{pmatrix} \xrightarrow{\text{Perspective division}} \begin{pmatrix} \frac{x}{-(z/d)} \\ \frac{y}{-(z/d)} \end{pmatrix} \longleftarrow \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{d} & 0 \end{pmatrix}\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \\ -\frac{z}{d} \end{pmatrix}$$