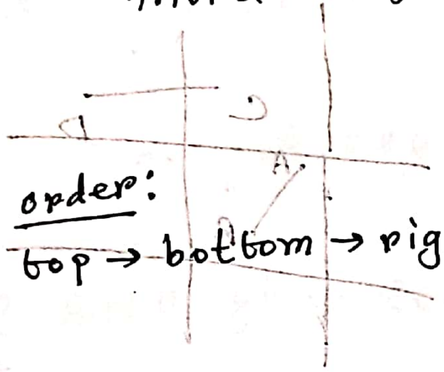
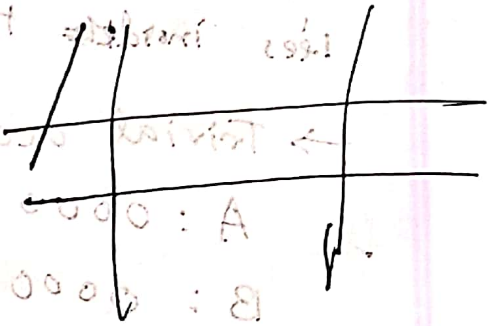
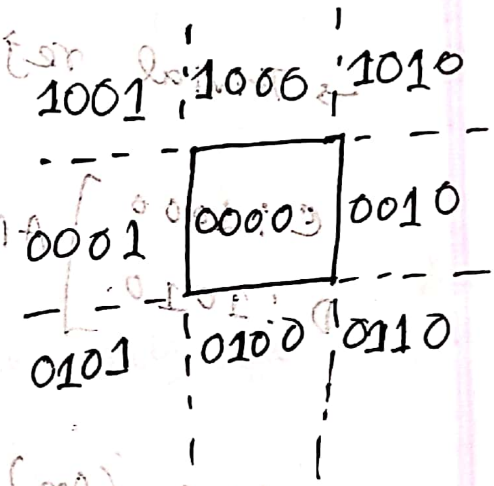
Clipping line

trivial acceptance.
trivial rejection.





order:
top → bottom → right → left.



→ First bit 1 if region above top edge, 0 otherwise.

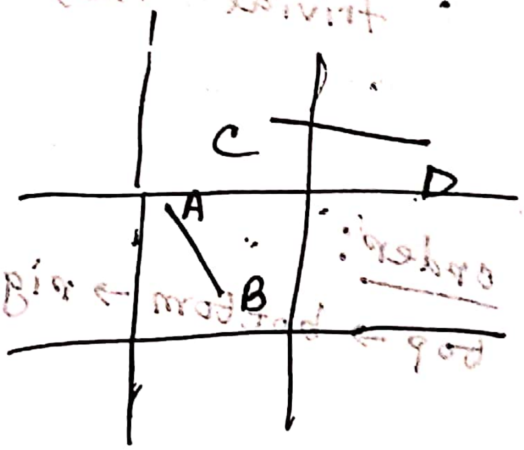
→ Second bit 1 if region below bottom edge, 0 otherwise.

→ Third bit if the right to right side, 0 otherwise.

→ Fourth bit if region is left to left bit, 0 otherwise.

→ If both endcodes are 0000 then it completely lies inside the clip rectangle.

→ Trivial acceptance



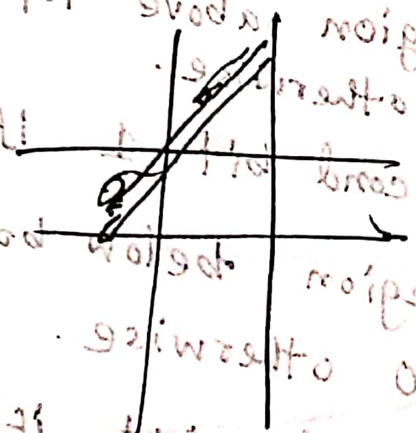
A : 0000

B : 0000

→ Trivial rejection

C : 1000 ⎤
D : 1010 ⎦ AND - gives 1000

→ above the top edge.



set bit : 1 (on)

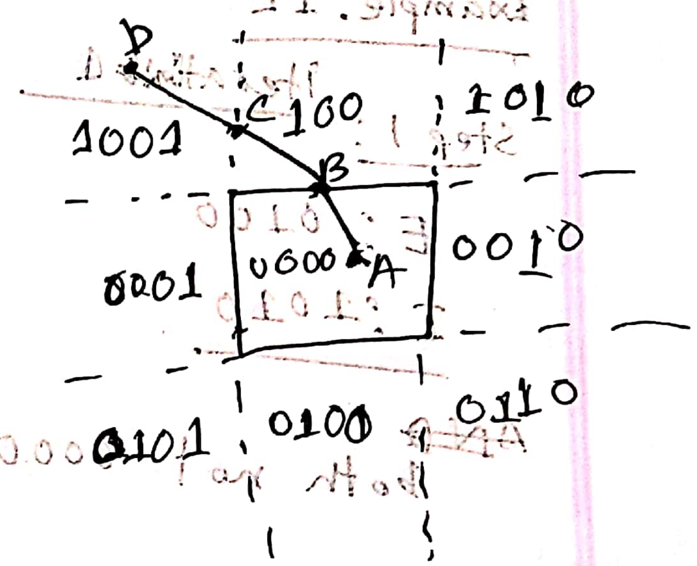
reset bit : 0 (off).

choose set-bit in the outcode

↳ ~~top edg~~

leftmost →

Example : Consider AD



1. A: 0000

D: 1001

Both not 0000.

2. AND gives zero
→ can't trivially rejected.

3. D: 1001.
→ B is found (Alphabetically we set/take point)

Assign B to 0000.

next Iteration:
Ite2:

1. A: 0000

B: 0000

Both endpoints are 0000

Example: IE

Iteration 1

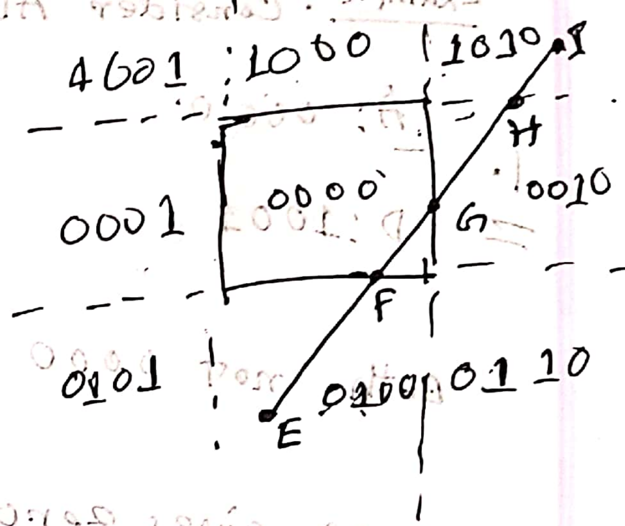
Step 1:

E: 0100

I: 1010

~~AND~~ 0000

both not 0000



Step 2:

AND gives zero.

Can't trivially rejected.

Step 3:

E(0100). Alphabetically

(F: 0000)

Iteration 2:

IF

1. I(1010)

F(0000)

Both not 00000

2. AND gives zero.

3. select outcode lies outside clip rectangle.

I = 1010.

new point H found (0010).

iteration 3:

1. F = 0000
H = 0010

both not 0000

2. AND gives zero

3. H = 0010, ~~new~~ leftmost set-bit : third bit = right edge for subdivision.

→ new point G is found:

G(0000)

iteration 4:

1. ~~A~~ F (0000)
G(0000)

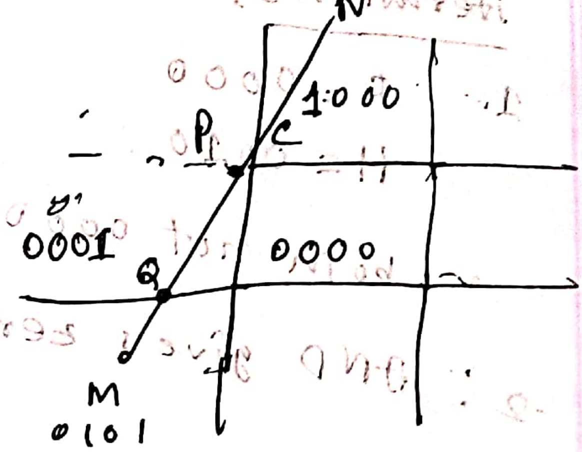
trivially accepted since both endpoints are 0000.

1) MN

2) QN.

3) QP.. → 0001, → 0001

↳ AND gives non-zero
trivially rejected.



works well two-cases:

→ very large clip region

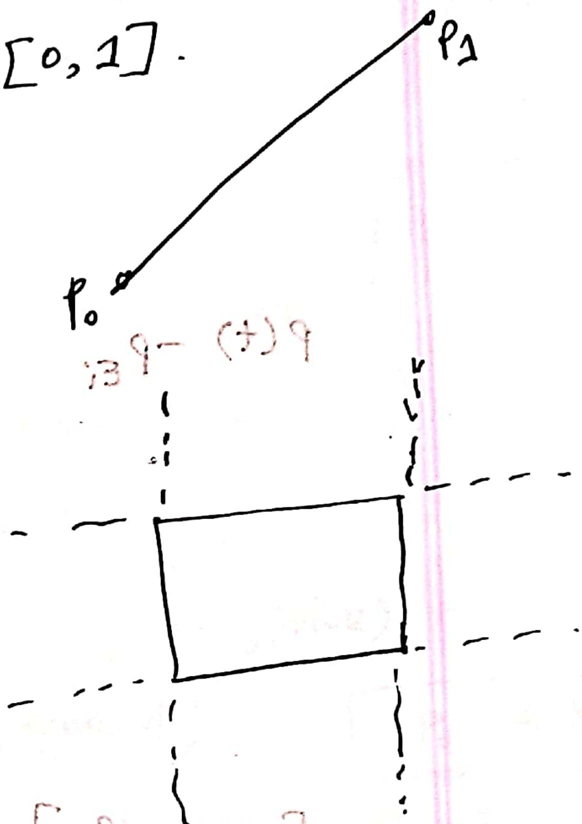
→ very small clip region

Lecture -2 Final Topic: Cyrus-Beck Lineclipping

Day: Saturday

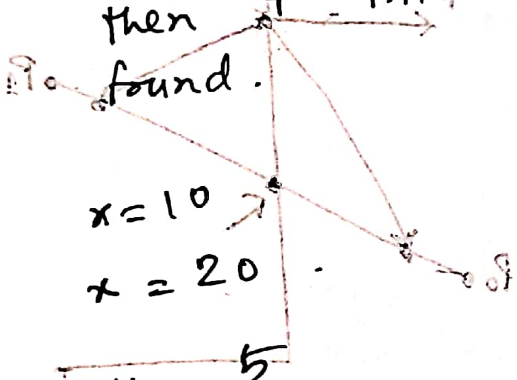
Cyrus-Beck Parametric line Clipping

$$P(t) = P_0 + (P_1 - P_0)t \; ; \; t = [0, 1].$$



* Extend clip edges.

not parallel then 4 intersection points found.

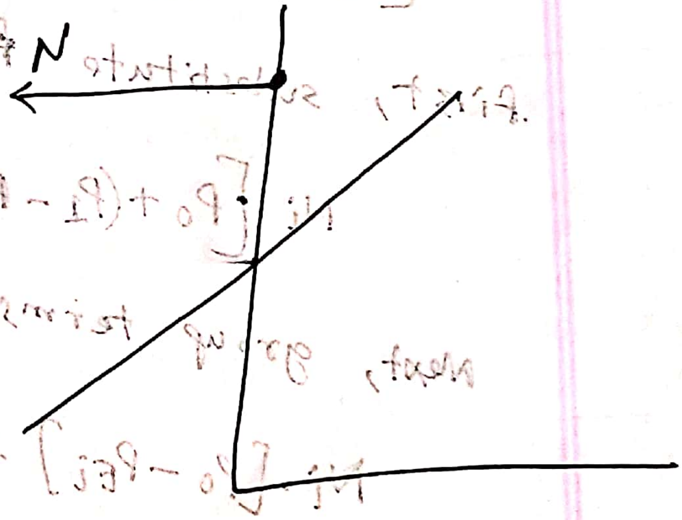


$x = 10$

$x = 20$

$y = 5$

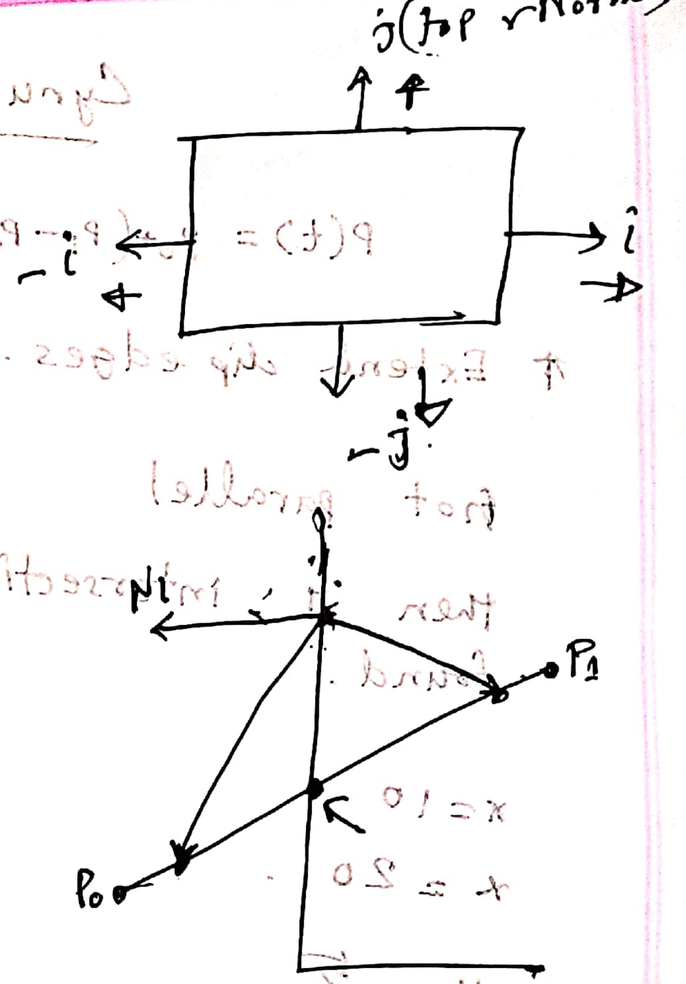
$y = 15$.



N = normal (লম্ব (রেখা))

		Normal
left:	$x = x_{min}$	$(-1, 0)$
right:	$x = x_{max}$	$(1, 0)$
bottom:	$y = y_{min}$	$(0, -1)$
top:	$y = y_{max}$.	$(0, 1)$

$P(t) - P_{Ei}$



$N_i \cdot [P(t) - P_{Ei}] = 0$.

first, substitute for $P(t)$:

$$N_i \left[P_0 + (P_1 - P_0)t - P_{Ei}\right] = 0$$

Next, group terms & distribute dot product.

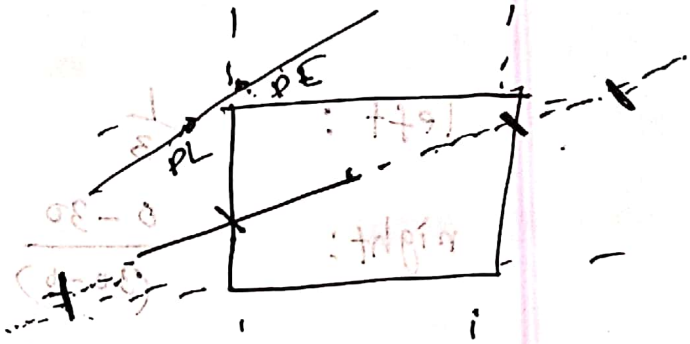
$$N_i \cdot [P_0 - P_{Ei}] + N_i [P_1 - P_0]t = 0$$

Let, $D = (P_1 - P_0)$ be vector from P_0 to P_1.

$$t = \frac{N_i [P_0 - P_{Ei}]}{-N_i \cdot D}$$

$\Rightarrow \boxed{\neq 0}$ (parallel না হলে)

Trivial rejection:



* $x_{min} = 10$

$x_{max} = 20$

$y_{min} = 5$

$y_{max} = 15$.

PE = potenti, entering. → $\boxed{N_i \cdot D} < 0$ (negative) (স্থূলকোণ)

PL = " leaving. → $N_i \cdot D > 0$ (positive) (সূক্ষ্মকোণ)

$(D = P_1 - P_0) = (x_1 - x_0, y_1 - y_0)$

$P_0 = (x_0, y_0)$

$P_1 = (x_1, y_1)$

$P_{Ei} = \cancel{(x_{min}, y)}$

	N_i	P_{Ei}	$N_i \cdot D$ ($\cancel{P_0 - P_{Ei}}$)	t
left:	$(-1, 0)$	(x_{min}, y)	$-(x_1 - x_0)$	$\frac{-(x_0 - x_{min})}{(x_1 - x_0)}$
right:	$(1, 0)$	(x_{max}, y)	$(x_1 - x_0)$	$\frac{(x_0 - x_{max})}{-(x_1 - x_0)}$
bottom:	$(0, -1)$	(x, y_{min})	$-(y_1 - y_0)$	$\frac{-(y_0 - y_{min})}{(y_1 - y_0)}$
top:	$(0, 1)$	(x, y_{max})	$(y_1 - y_0)$	$\frac{(y_0 - y_{max})}{-(y_1 - y_0)}$

left : $\frac{1}{3}$

right: $\frac{0-30}{(30-70)} = \frac{2}{3}$.

bottom $\frac{1}{6}$

top $\frac{1}{2}$

PE -30

PL 30

PE -30

PL 30

$\frac{1}{3}, \frac{1}{6} \rightarrow PE$.

$\frac{2}{3}, \frac{1}{2} \rightarrow PL$

* $tE = t$ value of PE having highest value.

$tE = \frac{1}{3}$.

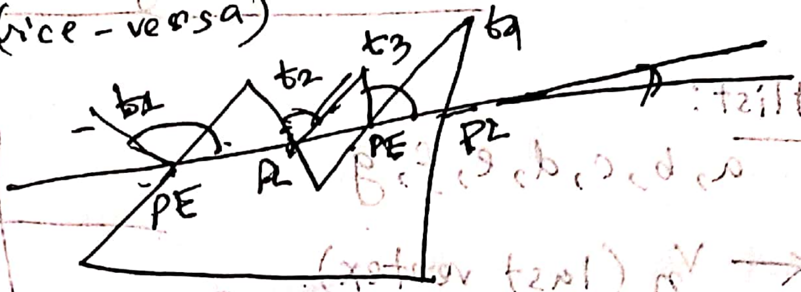
* $tL = t$ value of PL having lowest of t .

$tL = \frac{1}{2}$.

* $tE < tL$, accepted

$\frac{1}{3} < \frac{1}{2}$ (accepted).

convex: ভেতর থেকে বাহিরে (vice-versa)



$tE > tL$.

never works.

(concave polygon works there)

5*

		$N_i \cdot D$	t
left:	$x = 8$	$-(x_1 - x_0) = -20$ (PE)	$\frac{-(1-8)}{20} = \frac{7}{20}$
right:	$x = 18$	$(x_1 - x_0) = 20$ (PL)	$\frac{-(1-18)}{20} = \frac{17}{20}$
Bottom:	$x = 3$	$-(y_1 - y_0) = -20$ (PE)	$\frac{-(1-3)}{20} = \frac{1}{10}$
Top:	$x = 14$	$(y_1 - y_0) = 20$ (PL)	$\frac{-(1-14)}{20} = \frac{13}{20}$

$x_0 = 1$, $x_1 = 21$

$y_0 = 1$, $y_1 = 21$

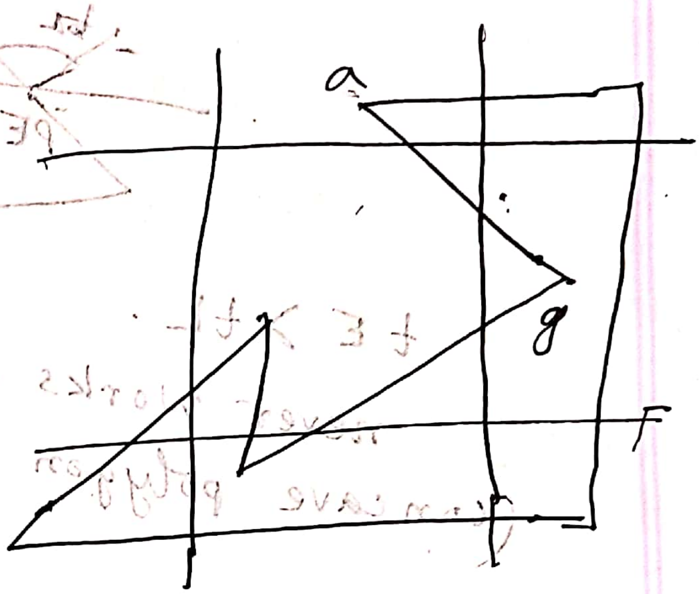
$tE = \frac{7}{20}$.

$tL = \frac{13}{20}$.

$tE < tL$. $\Rightarrow \frac{7}{20} < \frac{13}{20}$

(accepted)

Sutherland - Hodgson Polygon Clipping.



inputlist:
a, b, c, d, e, f, g

S ← V_n (last vertex).

P → first vertex. $\boxed{V_1}$

S → g

P = a

$\boxed{j \leftarrow 1}$

$\boxed{\text{Top, right, bottom, left.}}$

SP if both inside choose output P.

outside ~~both out~~ do nothing.

SP if " then Find intersect

S inside P outside point.

S outside P inside the find intersect point.

S P	int point	output	comm
g, a	1	1	g inside, a outside
a, b	–	–	both outside
b, c	2	2, c	b outside, c inside.

S.P	int	output	comments
c, d	—	d	both inside
d, e	—	e	both inside
e, f	—	f	both inside
f, g	—	g	both inside

1, 2, c, d, e, f, g → $\boxed{\text{output}}$

Right:

SP	int	output	comments
g, 1	3	3, 1	g outside, a inside.
1, 2	4	4	1 inside, 2 outside
2, c	—	—	2 outside, c outside
c, d	5	5, d	c outside, d inside.
d, e	—	e	both inside
e, f	—	f	both inside
f, g	6	6	~~both inside~~ f inside, g outside

output = 3, 1, 4, 5, d, e, f, 6

bottom: output of previous iteration = 3, 1, 4, 7, 8, e, 9, 10, 6

SP	int. point	output	comment
~~9,6~~ 6,3	—	3	both inside
3,1	—	1	both inside
1,4	—	4	both inside
4,5	7	7	4 inside, 5 outside
5,d	—	—	5 outside, d outside
d,e	8	8, e	d outside, e inside
e,f	9	9	e inside, f outside
f,6	10	10, 6	f outside, 6 inside

3, 1, 4, 7, 8, e, 9, 10, 6

left:

SP	int.	output	comments
6,3	—	3	both inside
3,1	—	1	both inside
1,4	—	4	both inside
4,7	—	7	both inside
7,8	11	11	~~both~~ 7 inside, 8 outside
8,e	12	12, e	8 outside, e inside

e, 9		9	both inside
9, 10	—	10	both inside
10, 6	—	6	both inside

output: 3, 1, 4, 7, 11, 12, e, 9, 10, 6.

Lecture - 04 Final Topic: Hidden-Surface

Hidden Surface Removal

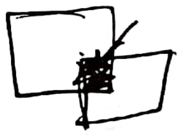
Visible Surface

i) Image Precision

ii) Object "

Invisible Primitives

Why Polygon can be invisible?



efficiency reason:
- ~~→ Outside f. of - view.~~
- ~~→ back facing~~
- → Polygon is occluded by object nearer the viewpoint

polygon when occluded .?

Conservative Algorithm :

Doesn't give final output.

Rest Algo → Img & obj. → এর accuracy বাড়াতে।

Works : back-face culling (বাদ দেওয়া), canonical view volume clipping, have to feed results.

Image Precision:

$O(n)$

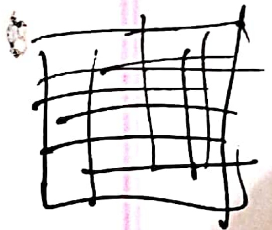
for each Pixels, determine closest object pierced by

the ray through the pixel (nearest point দিতে হবে).

draw the pixel in appro. color

* $O(np)$; p = # pixels.

40x 40



100X100

size diff

Brute-Force obj → all obj. checking.

* needs to redone if img resized

(for that its little inefficient)

(Ray-Tracing) works with this.

Object Precision:

$O(n^2)$

(Rasterization)

Brute-Force,

~~determine~~

$n << p$, so this seems good.

Tends to be slower in practice.

resolve for all possible view directions

from given eye-point.

If image is resized calculation need to be redone?

⇒ No. We don't need to calculate if we resize.

Z – Buffer – Image Precision

* Intensity Buffer 2D ⇒ 3 टा layer रहा.
  - → RGB buffer
  - → background color set
* Depth ("Z") Buffer
  - → depth of scene ~~initialized~~ at each pixel.
  - → far depth initialized.

polygons are arbitrary ordered.

* Initialize → Each z-buffer cell ⇒ max z value. (255
  - Each frame " " ⇒ backgro. color (white)

for each polygon:

compute $z(x, y)$,

$z(x, y) < z$ buffer then.

z buffer$(x, y) \Leftarrow z(x, y)$.

Pixel$(x, y) \Leftarrow$ color of polygon at (x, y).

CamScanner

255 white $\rightarrow$ 127 (store) color : grey

127 $\rightarrow$ 128 ✗

127 $\rightarrow$ 127 ✗

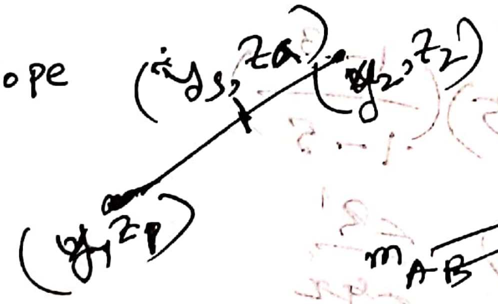
127 $\rightarrow$ 115 (store) color : black

255 $\rightarrow$ 115 (store) color : black.

z - values.

<u>Interpolation</u> :

slope (y_s, z_a) (y_2, z_2) (y_1, z_p)



$$z_a = z_1 - (z_1 - z_2)\frac{y_1 - y_s}{y_1 - y_2}$$

$$z_b = z_1 - (z_1 - z_3)\frac{y_1 - y_s}{y_1 - y_3}$$

$$z_p = z_b - (z_b - z_a)\frac{x_b - x_p}{x_b - x_a}$$

* Exercise!

$(x_1, y_1, z_1) = (1, 2, 3)$

$(x_2, y_2, z_2) = (10, 12, 20)$

* $y_s = 3.$

$(y_1, z_1) = (1, 3)$, $y_3, z_3 = (5, 5)$.

$(y_2, z_2) = (4, 4)$,

$$z_a = 3 - (3-4)\left(\frac{1-3}{1-4}\right)$$

$$= 3 - (-1)\frac{\cancel{-}2}{\cancel{-}3}$$

$$= 3 + \frac{2}{3}$$

$$= \frac{9+2}{3} = \frac{11}{3} = 3.67$$

* $\boxed{x_a = 1 \quad x_b = 10}$

$x_p = 3$

$$z_b = 3 - (3-5)\left(\frac{1-3}{1-5}\right)$$

$$= 3 - (-2)\frac{\cancel{-2}^{1}}{\cancel{-4}_{2}}$$

$$= 3 + 1 = 4.$$

$$z_p = 4 - (4-3.67)\frac{10-3}{10-1}$$

$$= 4 - (0.33)\frac{7}{9} = 3.74.$$

can be used for non-polygon surfaces.

Lower precision for higher-depth.

~~Low~~ High amount of memory required.

Lecture-05

Final

Topic-Rasterization

Day: Saturday.

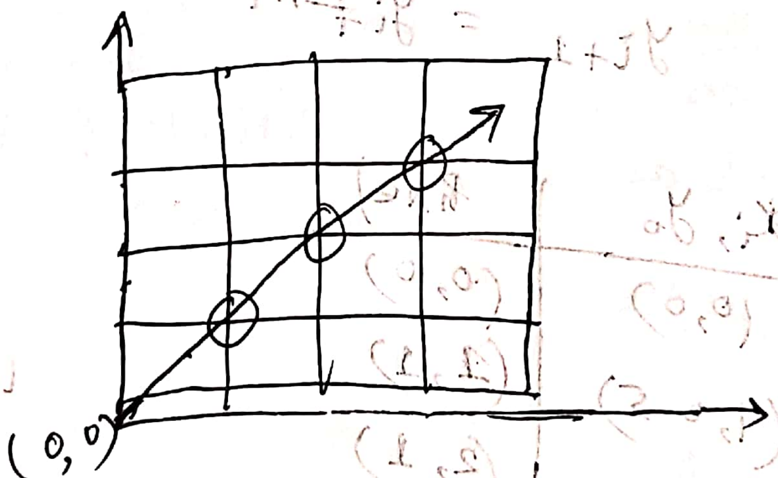
☐ Scan-line conversion

→ Pixels are represented as disjoint.

☐ Scan converting lines:

If the line has slope, $m = 1$.

Incremental along x and y-axis is 1.



Basic Increment Algorithm :- Line $P_0(4,5)$.

1. $m = 1$, $x_{i+1} = x_i + 1$.

$y_{i+1} = y_i + 1$.

$$m = \frac{\Delta y}{\Delta x} = \frac{8-5}{7-4} = \frac{3}{3} = 1$$

(x_1, y_1)	Pixel
	4,5
	5,6
	6,7
	7,8

2. $m < 1$, $x_{i+1} = x_i + 1$.

$y_{i+1} = y_i + m$

x_{i+1}

$y_{i+1} = round(y_{i+1})$

x_i, y_o	Pixel
(0,0)	(0,0)
(1,0.5)	(1,1)
	(2,1)
(2,1)	
(3,1.5)	(3,2)
(4,2)	(4,2)

Line 2: $P_0(0,0)$

$P_1(4,2)$

$$m = \frac{\Delta y}{\Delta x} = \frac{2-0}{4-0} = \frac{1}{2} = \boxed{0.5}$$

3. $m > 1$, $x_{i+1} = x_i + \frac{1}{m} \longrightarrow x_{i+1} = round(x_{i+1})$

$y_{i+1} = y_i + 1$.

Lecture-06

Topic: Rasterization

Final

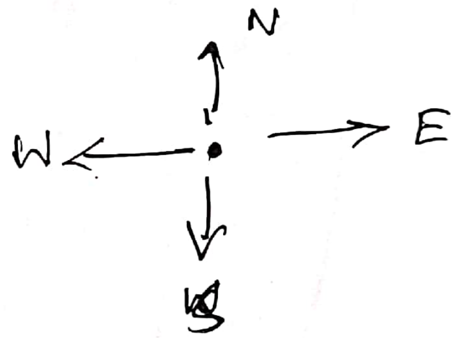
Date: 23-08-2022

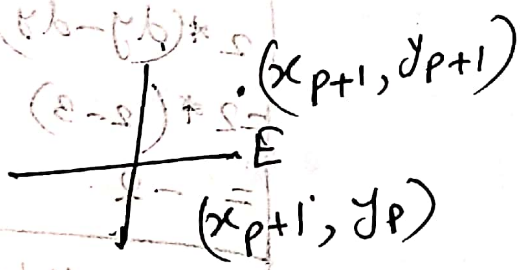
Day: Tuesday.

Midpoint-line Algorithm

$x_{i+1} = x_i + 1$

$y_{i+1} = y_i + m$





$y = mx + B$

$= \frac{dy}{dx} x + B$.

$\Rightarrow dx y = dy x + dx B$

$\Rightarrow (dy)x + (-dx)y + dx.B$

$= 0$

$f(x,y) = \cancel{(dy, -dx)}$

$ax + by + c$.

Algo works for $\Rightarrow 0 \leq m \leq 1$

Fig: 3.8

Line: $P_0\ (1, 2)$ $\quad P(4, 4)$

$$m = \frac{4-2}{4-1} = \frac{2}{3} < 1.$$

Decision	Calculation	Pixel
	$dx = x_1 - x_0$ $= 4 - 1 = \boxed{3}$ $dy = y_1 - y_0$ $= 4 - 2 = \boxed{2}$ $d = 2 * dy - dx$ $= 2 * 2 - 3$ $= \boxed{1}$ $incr\,E = 2 * dy$ $= 2 * 2$ $= \boxed{4}$ $incr\,NE =$ $2 * (dy - dy)$ $= 2 * (2 - 3)$ $= -2$	
$d > 0$	$d \leq 0$ (false) $d = 1 + (-2)$ $= \boxed{-1.}$ $x = 1 + 1 = 2$ $y = 2 + 1 = 3$	$(2, 3)$
$d < 0$	$d = -1 + (4)$ $= \boxed{3}.$ $x = 3,$	$(3, 3)$
$d > 0$	~~d =~~ $d = 3 - 2$ $= 1$	$(4, 4)$

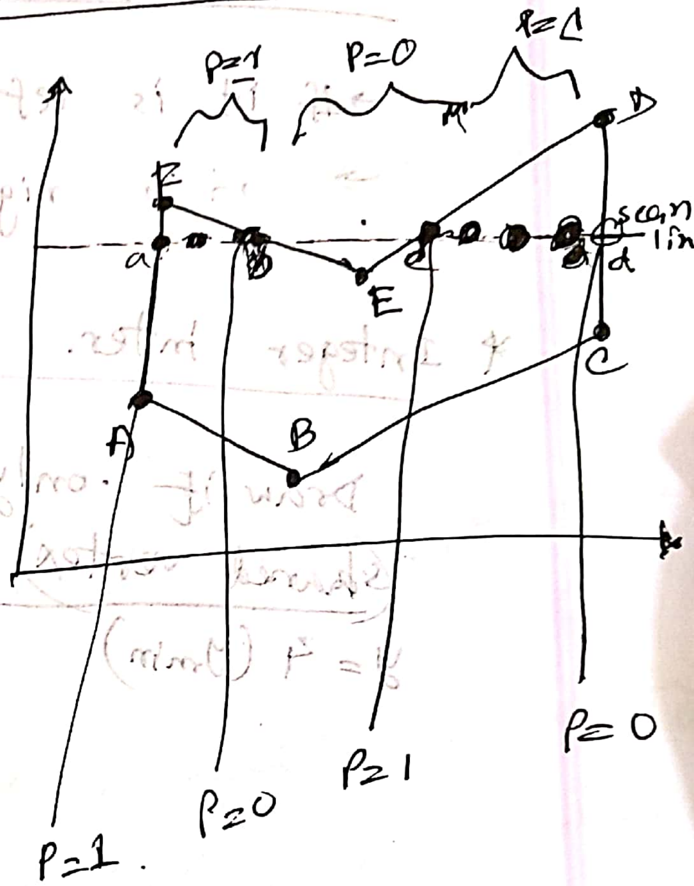
Day: Saturday
Lecture-07

Final

Topic: Rasterization

- Polygon Scan Conversion

Parity bit: 1 or 0.



i) scan line

ii) sort the interse

iii) use odd parity-bit = 1.

P = 0, even.

$\boxed{\text{only}}$ odd, then draw pixels.

4 steps /cases:

* Intersection point is fractional:-

→ If it is the left-most of the span outside polygon round up. (x right এ যাবে)

→ if it is the right-most of the span inside polygon round down (x left এ যাবে).

* Intersection point is integer:

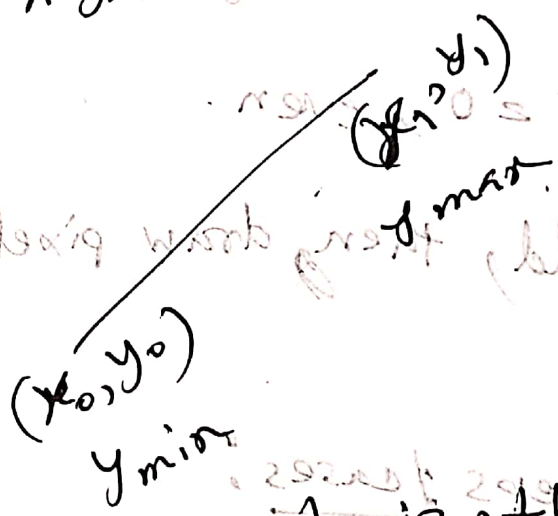
→ If it is left most — draw.

→ ,, ,, right ,, — don't draw.

* Integer inter. point shared vertex:

Draw if only when its y_{min} of an edge (shared vertex).

$y = 4$ (y_{min})



** Integer int. point is define horizontal edge.

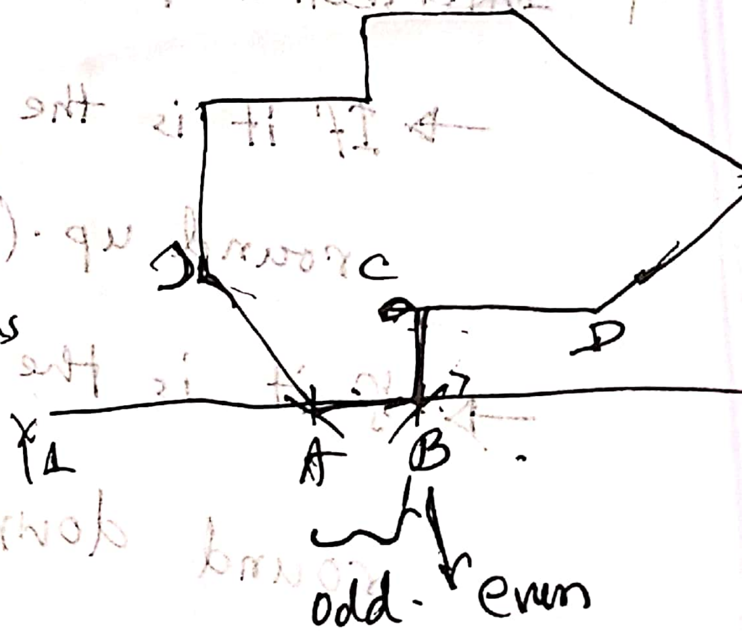
A is drawn. parity becomes odd

. AB drawn. parity ~~beco~~ remains odd.

B is y_{min} of CB.

parity becomes even.

draw stops



* → C is not Y_{min} of CD or BC. parity remains odd. CD drawn.

→ D is Y_{min} of DE. Parity becomes even. Drawing stops.

* → I not drawn.

I is not Y_{min} of IH or IJ. parity even

→ IH not drawn.

→ H is Y_{min} GH. H is considered parity odd. span up to next int. point is drawn

* → G is not Y_{min}. GF or GH. G is not considered, GF is not drawn.

<u>Summary</u>: Top edges not drawn.

Bottom " drawn.

Left edges are drawn

Right " are not drawn

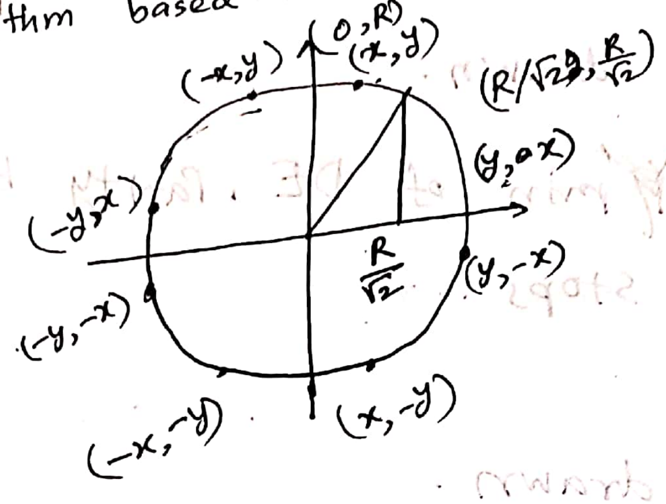
Lecture-08

Tuesday

Topic: Midpoint Circle (Rasterization)

Date:

Algorithm based on 2nd octant:



$P(x_p, y_p)$

$\rightarrow M(x_p+1, y_p-\frac{1}{2})$

$f(x,y) = x^2+y^2=R^2$

$d = f(M) = f(x_p+1, y_p-\frac{1}{2})$

positive $\rightarrow$ SE.

negative $\rightarrow$ E

on the circle/=0 $\Rightarrow$ (SE/E)

new midpoint $\rightarrow M_E$.

$d_{new} = M_E = (x_p+2, y_p-\frac{1}{2})$

$d_{new} = d_{old} + (2x_p+3)$

$= d_{old} + \Delta E$

* If a pixel in first octant (4,5) then calculate on other 7 octants:

1. $(y,x) = (4,5)$
2. $(x,y) = (5,4)$
3. $(-x,y) = (-5,4)$
4. $(-y,x) = (-4,5)$
5. $(-y,-x) = (-4,-5)$
6. $(-x,-y) = (-5,-4)$
7. $(x,-y) = (5,-4)$
8. $(y,-x) = (4,-5)$

* Calculate the pixels which gets rasterized in 2nd octant by using midpoint-circle algorithm by 2nd order difference for the following circle:

$$x^2 + y^2 = 5^2.$$

Condition	Calculation	Pixel	fourth
	$x=0, y=5$. $d = 1-5 = -4$ deltaE = 3 deltaSE = $-2*5+5$ $= -5$.	(0,5)	(-5,0)
$y > x$, T $d < 0$, T	~~circle points(0,5)~~ ~~d = 1-5~~ $d = -4+3$ $= -1$ deltaE = $3+2$ $= 5$ deltaSE = $-5+2$ $= -3$	(1,5)	(-5,1)

Condition	Calculation	Pixel	fourth
$y>0$, T $d<0$, T	$d=-1+5$ $=4$ deltaE $=5+2$ $=7$ deltaSE $=-3+2$ $=-1$	(2,5)	(-5,2)
$y>0$, T $d<0$, F	$d=4-1$ $=3$ deltaE $=7+2$ $=9$ deltaSE $=$ ~~5+4~~ $-1+4$ $=3$	(3,4)	(-4,3)
$y>0$, T $d<0$, F	$d=3+3$ $=6$ deltaE $=9+2$ $=11$ deltaSE $=3+4$ $=7$	(4,3)	(-3,4)
~~y>x~~ $y>x$, F	Stop		

Ray-Casting :-

$$R(x) = P_0 + t(P_1 - P_0)$$

↳ origin vector.

For every pixel construct a ray from the eye for every object in the scan find intersection with the ray keep in closet.

* This algorithm is slow.

Note on shading :-

* Surface / Scene characteristics :-
  - → surface normal
  - → direction to light
  - → Viewpoint.

* Material properties :
  - ↳ Diffuse (Matte)
  - ↳ Specular (Shiny)

Pinhole camera:

→ Box with ~~ty~~ tiny hole

→ Inverted Image

Simplified pinhole camera

→ Eye - image xy

remid (Frateuon)

Camera Description :-

→ Eye-point e (center)

Two-types of projection.

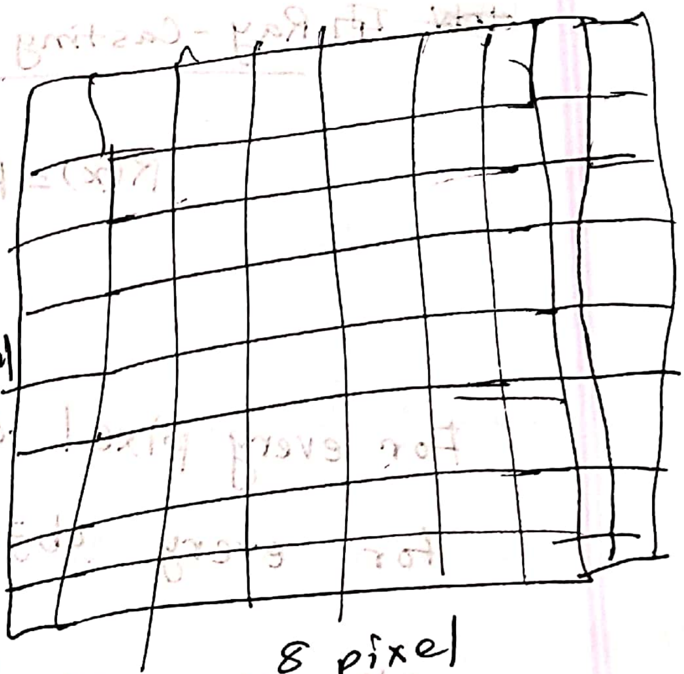
1) Perspective.

ii) Orthographic

→ Orthographic Camera

→ Ray Generation

8 pixel



8 pixel

* Calculation :-

center : (4,5,8)

$\overrightarrow{Horizontal} = \left(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right)$

$\overrightarrow{up} = \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0\right)$

size = 10 cm

origin (x, y) =

origin (5,5) $= (4,5,8) + (5-0.5) \times 10 \left(\frac{1}{\sqrt{3}} \times \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right)$.

Ray-Plane Intersection:

* Ray; $P(t) = R_o + tR_d$.

* plane, $H(P) = n \cdot P + d$

$n(R_o + tR_d) + D = 0$.

$$t = -\frac{n \cdot R_o + D}{n \cdot R_d}$$

$$P(t) = (0,1,5) + t\left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0\right).$$

$$H(P) = 3x + 4y + 5z - 10 = 0$$

$$\vec{n} = (3,4,5), \quad D = -10.$$

$$t = \frac{(0,1,5)(3,4,5) + (-10)}{-\frac{1}{\sqrt{2}}}.$$

$= 19\sqrt{2}$. (Ans.)

Additional t

1. $P(t) <$ tcurrent

2. $P(t) >$ tmin.

Question 4: Which Parameter will be stored finally using the ray casting method from the parameter $t_0 = 2$, $t_1 = 3$, $t_2 = -2$, $t_3 = 5$, $t_4 = 10$, $t_5 = 7$.

$t_{min} = 4$, $t_{current} = 3$.

$t > t_{min}$ $\qquad$ $t < t_{current}$.

$t_0 = 2$ ✗

$t_1 = 3$ ✗

$t_2 = -2$ ✗

$t_3 = 5$ ✓ $t_{next} = 5$.

Final

Topic: Ray sphere Intersection

Lecture-09

Date: 03.09.2022

$P(x, y, z)$

$x^2+y^2+z^2$

center = origin $= (0,0)$

$(x-\alpha)^2+(y-\beta)^2 = z^2$

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

[center = origin) $= (0,0,0)$

$\therefore (x-\alpha)^2+(y-\beta)^2+(z-\beta)^2 = r^2$

$H(P) = (P-C).(P-C) - r^2 = 0$

$P(t) = R_0 + tR_d$

$H(P) = P.P - r^2 = 0$

$P(t) = R_0 + tR_d \rightarrow$ [Ray eq^n]

$(R_0 + tR_d).(R_0 + tR_d) - r^2 = 0$

$ax^2+bx+c = 0$

$$x = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$$

$$t = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$$

$a = R_d.R_d$

$c = R_0.R_d - r^2$

$b = 2R_d.R_0$

$a = 1$

$b = 2 \cdot 0 = 0$

$c = 5 - 5^2 = -20$

$H(P) = P \cdot P^2 - 5^2 = 0$

$P(t) = (0,1,2) + t(-1,0,0)$

R_o R_d.

$$t = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$= \frac{-0 \pm \sqrt{0^2 - 4 \cdot 1 \cdot (-20)}}{2 \cdot 1}$$

$$= \pm \frac{4\sqrt{5}}{2}$$

[lowest value will be accepted]

$$= \pm 2\sqrt{5}$$

Accepted, $t = 2\sqrt{5}$

Geometric Ray Sphere Intersection:

Step 1: $R_o \cdot R_o = 5 < 5^2$.

Ray origin is inside sphere.

Step 2: $t_p = -R_o \cdot R_d$
$= 0$.

Step 3: $d^2 = R_o \cdot R_o - t_p^2$

$= 5 - 0$

$= 5$

$5 < 5^2$. (hit)

Step 4:

$t'^2 = r^2 - d^2$

$= 5^2 - 5$

$= 25 - 5$

$= 20$

$\therefore t' = \sqrt{20}$.

$\therefore t' = 2\sqrt{5}$.

$\therefore$ origin inside, $t = t_p + t'$

$= 0 + 2\sqrt{5}$

$= 2\sqrt{5}$

$=$ Accepted.

(Ans.)

normal $= \dfrac{Q}{|Q|}$ $\quad Q = P(t)$.

Topic: Ray Casting

Barycentric Definition of a Plane

" " of a Triangle

"

Triangle : $A = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}$

* eq^n

* Cramer's Rule:

Ray:

$R_o = (5, 6, 9)$

$R_d = (0, -1, 0)$

Triangle:

$a = (0, -5, 40)$

$b = (-10, 5, 90)$

$c = (40, 2, -5)$.

$P(t) = R_o + t R_d$

$$\begin{bmatrix} 0-(-10) & 0-40 & 0 \\ -5-5 & -5-2 & -1 \\ 40-90 & 40-(-5) & 0 \end{bmatrix} \begin{bmatrix} \beta \\ \gamma \\ z \end{bmatrix} = \begin{bmatrix} 0-5 \\ -5-6 \\ 40-9 \end{bmatrix}$$

$$\therefore \begin{bmatrix} \beta \\ \gamma \\ z \end{bmatrix} = \begin{bmatrix} -203/310 \\ -6/155 \\ 270/155 \end{bmatrix}$$

$\beta > 0 \leftarrow$ False

$\gamma > 0 \leftarrow$ False

$\beta + \cancel{\gamma} < 1 \leftarrow$ True.

$\vec{n} = (b-a) \times (c-a)$

$= (a-b) \times (a-c)$.

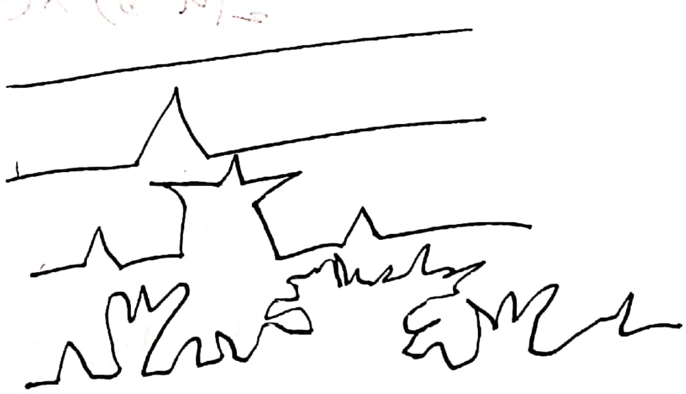
Lecture - 11 Final Day: Tuesday
Date: 13-09-2022

Topic: Fractals.

→ Koch Curve
→ Koch Snowflake
→ Dragon Curve
→ Hilbert "

* implemented using recursive functions.

* Koch Curve.



Golden ratio

Gen length = 1.

Gen 1 length = $4 \times \frac{1}{3} = \frac{4}{3}$

Gen 2 " = $\frac{4}{3} \times \frac{4}{3} = \left(\frac{4}{3}\right)^2$

How to draw Koch curve Gen, $n = \left(\frac{4}{3}\right)^n$.

if $l \to \infty$, length $= (4/3)^{\infty}$

$$= (1.33)^{\infty}$$

$$= \infty$$

* Koch Snowflake Curve:

* Perimeter Koch Snowflake Curve.

* Area of Koch Snow flake.

Area of $S_0 = a_0$.

Area of $S_1 = a_0 + 3(a_0/9)$

$$= a_0 + a_0/3.$$