CSE423

Name: Shihab Muhtasim

ID: 21301610

sec : 8 (AJA)

Ans to Q 1

(a)

I would choose midpoint line algorithm to draw a line. In DDA algorithm we have to round up the slop and add it to corresponding x or y. Then again we have to round it off to an integer. Hence, It is less accurate. Since we have to deal with floating points its not efficient as well.

However, midpoint algorithm overcomes these issues and procedes without any float operations or rounding errors.

Ans to qr 1(b)

$y = -2.5x + 10$

intersect y axis, $x = 0$

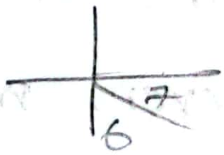
$y = 10 \quad \therefore A = (0, 10)$

intersect x axis, $y = 0$

$x = \frac{10}{2.5} = 4 \quad B = (4, 0)$

zone : $dx = 4$, $dy = -10$

$\frac{|dy|}{|dx|} > 1$

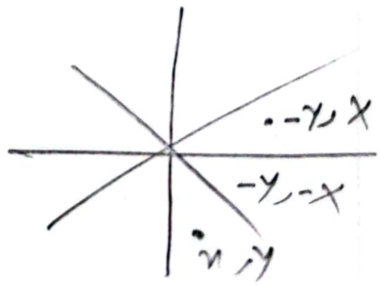


$\therefore$ zone = 6

Essential derivatives for ploting :

First convert these to zone 0

$A(-y, x) = (-10, 0)$



$B(-y, x) = (0, 4)$

$dy = 4$

$dx = 10$

$d_{init} = 2dy - dx = -2$

$\Delta NE = 2(dy - dx) = -12$

$\Delta E = 2dy = 8$

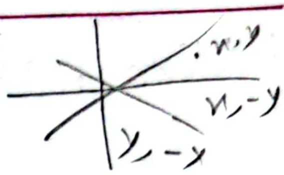
$d_{NE} = d + \Delta NE$

$d_E = d + \Delta E$

Ans to 1(c)

A(-10,0) B(0,4)

↳ in zone 0



$d = 2dy - dx = -2$

$dy = 4$
$dx = 10$

$\Delta NE = 2(dy - dx) = -12$ $\qquad$ $\Delta E = 2dy = 8$

	zone 0 X	z0 y	d	d'	status of Direction	z0 x	z0 y	z6 x'=y	z6 y'=-x
1	-10	0	-2	6	E	-10	0	0	10
2	-9	0	6	-6	NE	-9	0	0	9
3	-8	1	-6	2	E	-8	1	1	8
4	-7	1	2	-10	NE	-7	1	1	7
5	-6	2	-10	-2	E	-6	2	2	6
6	-5	2	-2	6	E	-5	2	2	5
7	-4	2	6	-6	NE	-4	2	2	4

Ans. to qr 2(a)

start $(0, p)$

E 10 times $= (10, p)$

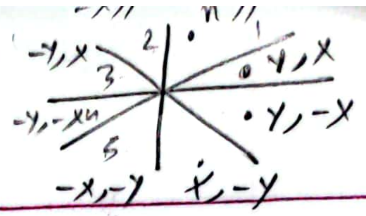
South 6 times $= (10+6, p-6)$

$= (16, p-6)$

Ans to qr 2(b)

The value of dinit $= 1.25 - r$ causes efficiency problems as it makes all the later d values to be floating point numbers. However, we can solve it by replacing it with $1-r$, becausse using $1-r$ shows same characteristics as $1.25-r$. Only in case $r=1$, $1.25-r$ gives 0.25 and $1-r$ gives 0, in other cases all reading gives pos or neg vals in either of thes operations.

Ans to qr 2(c)



Given radius $r = 9$

$d = 1 - r = 1 - 9 = -8$

$d_{SE} = d + 2x - 2y + 5 \quad d_E = d + 2x + 3$

x	y	d	status	d'	Z5 X = -X	Z5 Y = -Y	Z5+C X	Z5+C Y
0	9	-8	E	-8+0+3 = -5	0	-9	-4	4
1	9	-5	E	-5+2+3 = 0	-1	-9	-5	4
2	9	0	SE	0+4-18+5 = -9	-2	-9	-6	4
3	8	-9	E	-9+6+3 = 0	-3	-8	-7	5
4	8	0	SE	0+8-16+5 = -3	-4	-8	-8	5
5	7	-3	E	-3+10+3 = 10	-5	-7	-9	6
6	7	10	SE	10+12-14 +5 = 13	-6	-7	-10	6
7	6	13	SE	stops	-7	-6		

Ans to 3(a)

```
def calculate outcode (x,y):
① if x < xmin :
        bit 0 = 1
② if x > xmax :
        bit 1 = 1
③ if y < ymin :
        bit 2 = 1
④ if y > ymax :
        bit 3 = 1
```

Ans to Q. 3(b)

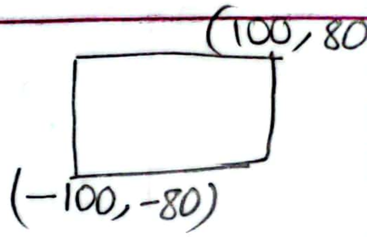
At most 6 clippings are done while clipping a 3D Line using the cohen-sutherland line clipping algorithm.

Names:

1. Intersection with top
2. Intersection with botton
3. Intersection with Right
4. Intersection with left
5. Intersection with far
6. Intersection with near

8

Ans to qr no 3(c)



(−160, 90) to (150, −88)

$D = P_1 - P_0 = (310, -178)$

Let, $t_E = 0$,
$t_L = 1$

Boundary	N	N.D	t	PE/PL	t_E	t_L
Left	−1, 0	−310	$\frac{-(-160 - -100)}{150 - -160} = 0.19$	PE	0.19	1
Right	1, 0	310	0.83	PL	0.19	0.83
Above	0, 1	−178	0.05	PE	0.19	0.83
below	0, −1	178	0.95	PL	0.19 t_E max	0.83 t_L min

$$t_L = \frac{-(x_0 - x_{min})}{x_1 - x_0} \qquad t_R = \frac{-(x_0 - x_{max})}{x_1 - x_0}$$

$$t_B = \frac{-(y_0 - y_{min})}{y_1 - y_0} \qquad t_A = \frac{-(y_0 - y_{max})}{y_1 - y_0}$$

$$P(t) = P_0 + t(P_1 - P_0)$$

$$P(t) = (-160, 90) + t(310, -178)$$

$$= -160 + t310,\ 90 - 178t$$

$$P(0{\cdot}19) = -101{\cdot}1,\ 56{\cdot}18$$

$$P(0{\cdot}83) = 97{\cdot}3,\ -57{\cdot}74$$