

### Experiment-3

Aim: To study the difference between Digital Differential Analyser (DDA) and Bresenham Line drawing Algorithm.

## Theory

- DDA Line Drawing Algorithm is an incremental scan conversion method to determine points on screen to draw a line.
- Bresenham Line Drawing Algorithm is an accurate, efficient raster line drawing algorithm that scan converts line using only incremental integer calculations.
- In Bresenham's algorithm, the next point to be plotted with respect to the value of decision parameter and increment will be always integer.
- Floating point arithmetic in DDA algorithm is time-consuming, which results in poor end point accuracy of a line segment. The use of floor/ceil function may increase the accuracy of the point.
- Bresenham algorithm can work efficiently and provides more end point accuracy for the gentle slope lines and sharp slope line without having floating calculation.
- DDA is the simplest algorithm and does not require special skills for its implementation.

## Procedure

### DDA ALGORITHM :

---

1. Start.
2. Accept End points of the line which are  $(x_1, y_1)$  and  $(x_2, y_2)$ .
3. Calculate the Length of the Line segment:

```
if ( abs(x2-x1) >= abs(y2-y1) )  
    then  
        length = abs(x2-x1).  
    else  
        length = abs (y2-y1).
```

4. Calculate the increment in X and Y directions respectively as:

```
xincr = (x2-x1) / length.  
  
yincr = (y2-y1) / length.
```

5. Initialize:

```
x=x1+0.5.  
y=y1+0.5.
```

```
i=1.
```

6. while ( i <= length )

```
{  
  
    plot (integer(x), integer(y))  
  
    x = x + xincr.
```

```
y = y + yincr.  
  
i   = i + 1.  
  
}
```

7. Stop.

---

## BRESENHAM ALGORITHM :

---

Let considering the line segment which has Gentle Slope

Assume:  $x_2 > x_1$  and  $y_2 > y_1$ .

1. Start.
2. Accept end points of a line segment  $(x_1, y_1), (x_2, y_2)$ .

3. Calculate:

```
4. dx = abs(x2-x1)  
5.   dy = abs(y2-y1)
```

4. Calculate the Length of the Line segment as:

```
if ( dx >= dy )
```

```
then  
  
length = dx  
  
else  
  
length = dy
```

5. Calculate Initial Decision Parameter as:

```
d = 2 * dy - dx
```

6. Calculate:

```
7. incr1 = 2 * dy - dx  
8.      incr2 = 2 * dy
```

7. Calculate xincr as:

```
8. If ( x2 > x1 )  
9.  
10. then  
11.  
12. yincr = 1  
13.  
14. else  
15.      yincr = -1
```

8. Initialize:

```
9. x=x1  
10.  
11. y=y1  
12.      i=1
```

13. while(i <= length)

```
14. {  
15.  
16. plotpoint (x ,y)  
17.  
18. if(d >= 0)  
19.  
20. then  
21.  
22. x = x + xincr  
23.  
24. y = y + yincr  
25.  
26. d = d + incr1
```

```

27.
28. else
29.
30. x = x + xincr
31.
32. d = d + incr2
33.
34. i++
35.
    }

```

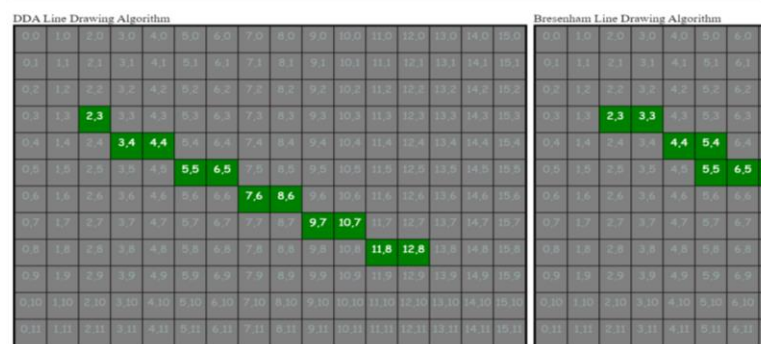
36. Stop.

Stimulation:

Draw a line from (2,3) to (12,8) using DDA Line Drawing Algorithm AND Bresenham Line Drawing Algorithm

Q1 Q2 Q3

Check Co-ordinates Clear Canvas



Conclusion: From this we can find the difference between DDA Algorithm and Bresenham Algorithm.