

Name: Shaikh Salif Aminuddin

CLASS: S2 **ROLL NO:** 2201094

EXPERIMENT NO: 1

AIM: Implement DDA Line Drawing algorithm (dotted/dashed/thick)

THEORY: DDA stands for Digital Differential Analyzer. It is an incremental method of scan conversion of line. In this method calculation is performed at each step but by using results of previous steps.

Advantage:

- It is a faster method than method of using direct use of line equation.
- This method gives overflow indication when a point is repositioned.
- It is an easy method because each step involves just two additions.

Disadvantage:

- It involves floating point additions rounding off is done. Accumulations of round off error cause accumulation of error.
- Rounding off operations and floating point operations consumes a lot of time.
- It is more suitable for generating line using the software. But it is less suited for hardware implementation.

DDA Algorithm:

Step1: Start Algorithm

Step2: Declare x_1 , y_1 , x_2 , y_2 , dx , dy , x , y as integer variables.

Step3: Enter value of x_1 , y_1 , x_2 , y_2 .

Step4: Calculate $dx = x_2 - x_1$

Step5: Calculate $dy = y_2 - y_1$

Step6: If $ABS > ABS(dy)$

Then $step = abs(dx)$

Else

Step7: $x_{inc} = dx/step$

$y_{inc} = dy/step$

assign $x = x_1$

assign $y = y_1$

Step8: Set pixel (x, y)

Step9: $x = x + x_{inc}$

$y = y + y_{inc}$

Set pixels (Round (x), Round (y))

Step10: Repeat step 9 until $x = x_2$

Step11: End Algorithm

CODE:

```
#include<graphics.h>
```

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
void main()
```

```
{
```

```
intgd = DETECT,gm, i; float x, y,dx,dy,steps;
```

```
int x0, x1, y0, y1;
```

```
initgraph(&gd, &gm, "C:\\TURBOC3\\BGI");
```

```
setbkcolor(WHITE);
```

```
x0 = 100, y0=200, x1 = 500, y1 = 300;
```

```
dx = (float)(x1 - x0);
```

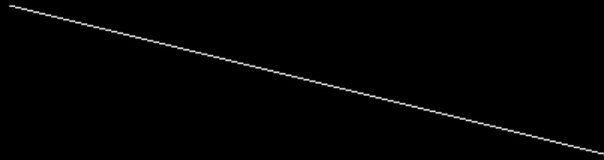
```

dy= (float)(y1 - y0);
if(dx>=dy) {
steps = dx;
} else{
steps = dy;
}
dx = dx/steps;
dy=dy/steps;
x = x0;
y=y0; i=1;
while(i<= steps)
{
    putpixel(x, y, RED); delay (30);
    x += dx;
    y += dy;
    i=i+1;
}
getch();
closegraph();
}

```

OUTPUT:

Name= S&LIF
Roll no.= 94



CONCLUSION: We learned about how to implement DDA Line Drawing algorithm