

COMPUTER GRAPHICS AND MULTIMEDIA

LAB ASSIGNMENT 1

By: Sparsh Arya

Registration number: 17BEC0656

Slot: L3+L4

DDA Line Drawing Algorithm

Algorithm

Step 1:

Read the input of the 2 end points of the line as (x_1, y_1) & (x_2, y_2) such that $x_1 \neq x_2$ and $y_1 \neq y_2$

Step 2:

Calculate $dx = x_2 - x_1$ and $dy = y_2 - y_1$

Step 3:

if($dx \geq dy$)

step= dx

else

step= dy

Step 4:

$x_{in} = dx / \text{step}$ & $y_{in} = dy / \text{step}$

Step 5:

$x = x_1 + 0.5$ & $y = y_1 + 0.5$

Step 6:

```
for(k = 0; k < step; k++)
```

```
{
```

```
    x = x + xin
```

```
    y = y + yin
```

```
    putpixel(x, y)
```

```
}
```

Code

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

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

```
#include <math.h>
```

```
#include <dos.h>
```

```
void main( )
```

```
{
```

```
    float x,y,x1,y1,x2,y2,dx,dy,step;
```

```
    int i,gd=DETECT,gm;
```

```
    initgraph(&gd,&gm,"c:\\turbo3\\bgi");
```

```
    printf("Enter the value of x1 and y1 : ");
```

```
    scanf("%f%f",&x1,&y1);
```

```
    printf("Enter the value of x2 and y2: ");
```

```
    scanf("%f%f",&x2,&y2);
```

```
    dx=abs(x2-x1);
```

```
    dy=abs(y2-y1);
```

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

```
        step=dx;
```

```
    else
```

```
        step=dy;

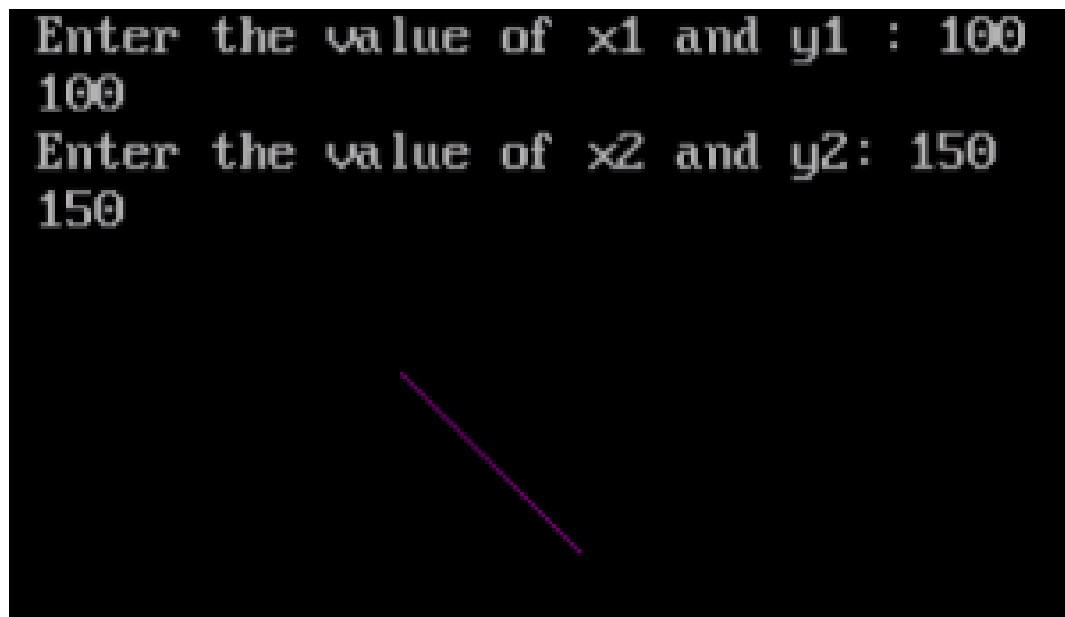
    dx=dx/step;
    dy=dy/step;

    x=x1;
    y=y1;

    i=1;
    while(i<=step)
    {
        putpixel(x,y,5);
        x=x+dx;
        y=y+dy;
        i=i+1;
        delay(100);
    }

    closegraph();
}
```

OUTPUT



Bresenham's Line Drawing Algorithm

Algorithm

Step 1:

Read the input of the 2 end points of the line as (x_1, y_1) & (x_2, y_2) such that $x_1 \neq x_2$ and $y_1 \neq y_2$

Step 2:

$$dx = \text{abs}(x_a - x_b)$$

$$dy = \text{abs}(y_a - y_b)$$

Step 3:

Initialize p as a variable whose value is defined as below.

$$p = 2 * dy - dx$$

Step 4:

Use the following conditions to initialize xend.

```
if xa > xb
{
    x=xb
    y=yb
    xend=xa
}
else
{
    x = xa
    y = ya
    xend=xb
}
setpixel (x,y)
```

Step 5:

To the values on the graph we use the following rules....

```
while x < xend
{
    x=x+1
    if (p < 0)
    {
        p=p+2 * dy
```

```
}  
  
else  
  
{  
  
y=y+1  
p=p+2*(dy-dx)  
  
}  
  
setpixel(x,y)  
  
}
```

Code

```
#include<stdio.h>  
  
#include<graphics.h>  
  
  
void drawline(int x0, int y0, int x1, int y1)  
{  
  
    int dx, dy, p, x, y;  
  
  
    dx=x1-x0;  
    dy=y1-y0;  
  
  
    x=x0;  
    y=y0;
```

```

p=2*dy-dx;

while(x<x1)
{
    if(p>=0)
    {
        putpixel(x,y,7);
        y=y+1;
        p=p+2*dy-2*dx;
    }
    else
    {
        putpixel(x,y,7);
        p=p+2*dy;
    }
    x=x+1;
}

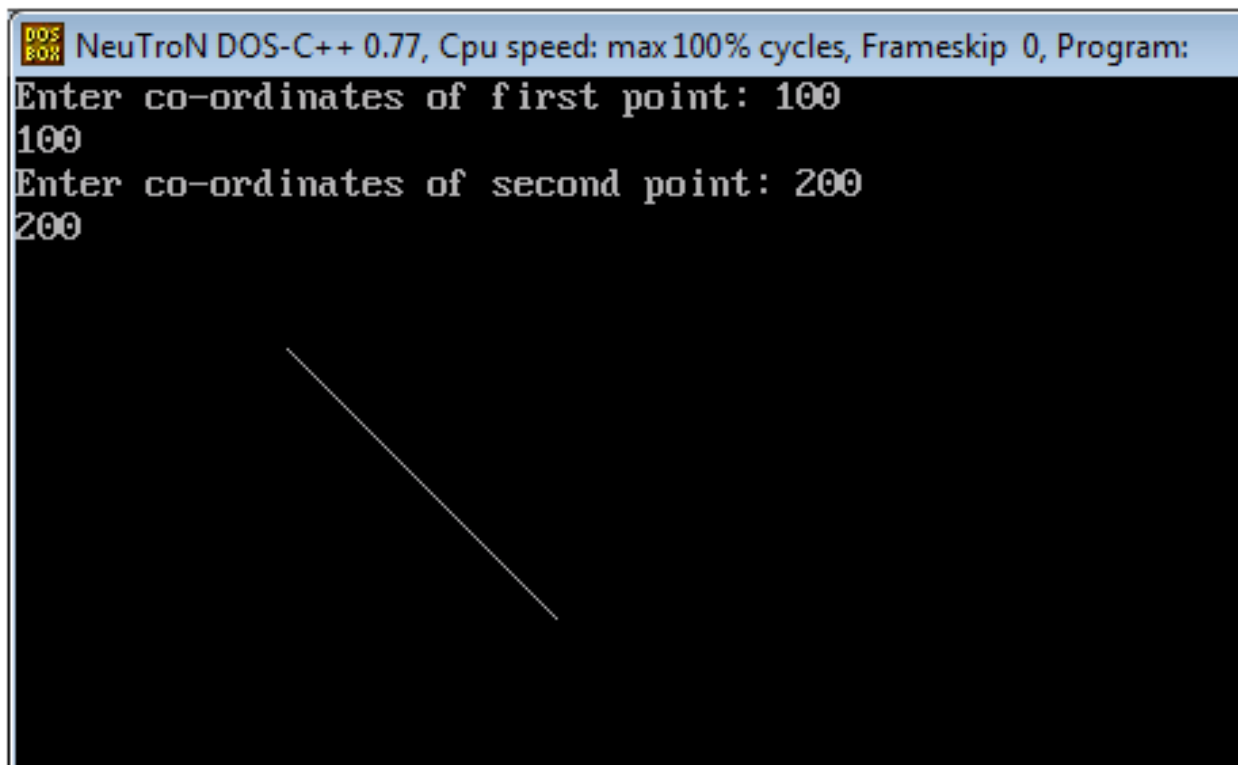
int main()
{
    int gdriver=DETECT, gmode, error, x0, y0, x1, y1;
    initgraph(&gdriver, &gmode, "c:\\turbo3\\bgi");

```



```
printf("Enter co-ordinates of first point: ");  
  
scanf("%d%d", &x0, &y0);  
  
printf("Enter co-ordinates of second point: ");  
  
scanf("%d%d", &x1, &y1);  
  
drawline(x0, y0, x1, y1);  
  
return 0;  
  
}
```

Output



```
NeuTroN DOS-C++ 0.77, Cpu speed: max 100% cycles, Frameskip 0, Program:  
Enter co-ordinates of first point: 100  
100  
Enter co-ordinates of second point: 200  
200  
  
A line is drawn from the point (100, 100) to the point (200, 200).
```