

<b>Name</b>	<b>:</b>	<b>V Vikram</b>	<b>Class</b>	<b>:</b>	<b>CSE 'C'</b>
<b>Reg. No.</b>	<b>:</b>	<b>18 5001 194</b>	<b>Date</b>	<b>:</b>	<b>04/08/2021</b>
<b>Subject</b>	<b>:</b>	<b>UCS1712---Graphics and Multimedia Lab</b>			

## **QUESTION :**

### **Lab Exercise 2 :**

#### **DDA Line Drawing Algorithm in C++ using OpenGL**

To plot points that make up the line with endpoints (x0,y0) and (xn,yn) using the DDA line drawing algorithm.

**Case 1** : +ve slope Left to Right line

**Case 2** : +ve slope Right to Left line

**Case 3** : -ve slope Left to Right line

**Case 4** : -ve slope Right to Left line

Each case has two subdivisions

(i)  $|m| \leq 1$  (ii)  $|m| > 1$

## **CODE :**

```
#include<GL/glut.h>
#include<stdlib.h>
#include<iostream>
#include<windows.h>
#include<GL/GL.h>
#include<GL/GLU.h>
#include<stdio.h>
#include<math.h>

float x1, x2, m, y2;

void init(void)
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glClearColor(0.0, 0.0, 0.0, 0.0);
```

```

        glColor3f(255.0f / 255.0f, 255.0f / 255.0f, 255.0f / 255.0f);
        glPointSize(2.0f);
        glEnable(GL_DEPTH_TEST);
        glMatrixMode(GL_PROJECTION);
        glLoadIdentity();
        gluOrtho2D(-100, 100, -100, 100);
    }
    void display(void)
    {
        float dy, dx, step, x, y, k, Xin, Yin;
        dx = x2 - x1;
        dy = y2 - m;

        if (abs(dx) > abs(dy))
        {
            step = abs(dx);
        }
        else
            step = abs(dy);

        Xin = dx / step;
        Yin = dy / step;

        x = x1;
        y = m;
        glBegin(GL_POINTS);
        glVertex2i(x, y);
        printf("m=%f", dy/dx);
        printf("\n(%.2f,%.2f)", x, y);
        glEnd();

        for (k = 1; k <= step; k++)
        {
            x = x + Xin;
            y = y + Yin;

            glBegin(GL_POINTS);
            glVertex2i(x, y);
            printf("\n(%f,%f)", x, y);
            glEnd();
        }

        glFlush();
    }

int main(int argc, char** argv) {
    printf("Enter x1 : ");
    scanf_s("%f", &x1);
    printf("Enter y1 : ");
    scanf_s("%f", &y1);

```

```
printf("Enter x2 : ");
scanf_s("%f", &x2);
printf("Enter y2 : ");
scanf_s("%f", &y2);

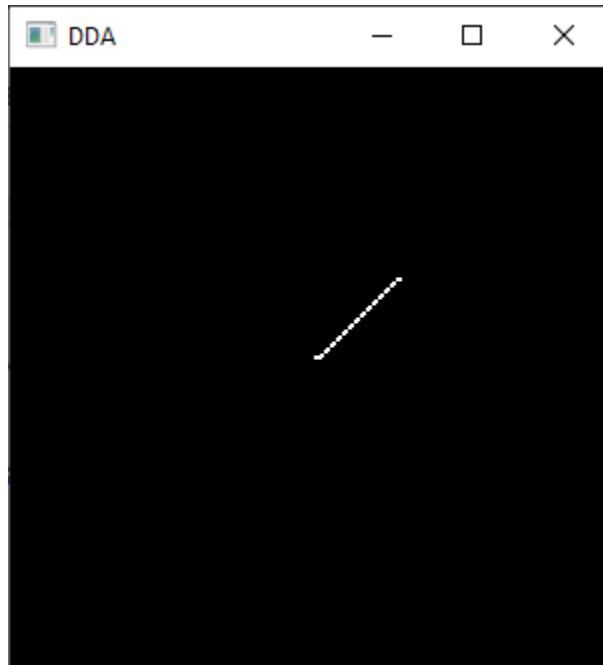
glutInit(&argc, argv);
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize(300, 300);
glutInitWindowPosition(100, 100);
glutCreateWindow("DDA");
init();
glutDisplayFunc(display);
glutMainLoop();

return 0;
}
```

## OUTPUT SNAPSHOTS :-

+ve slope left to right:

$$m \leq 1$$

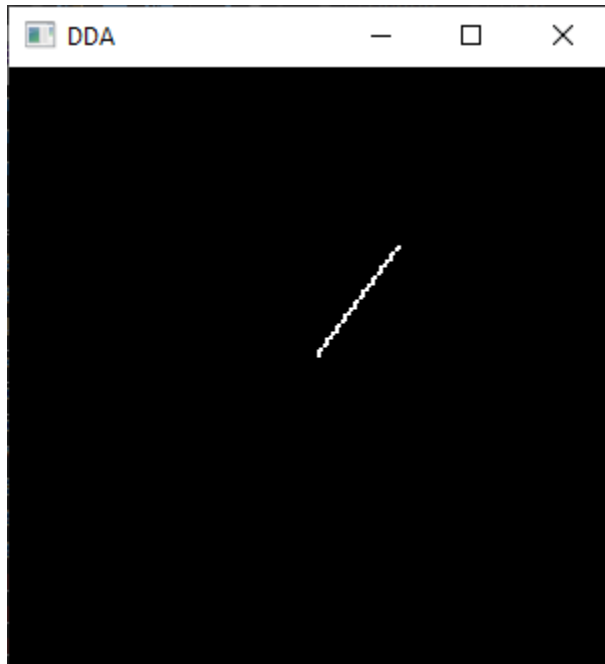


C:\Vikram\Vikram\_SEM-7\Graph

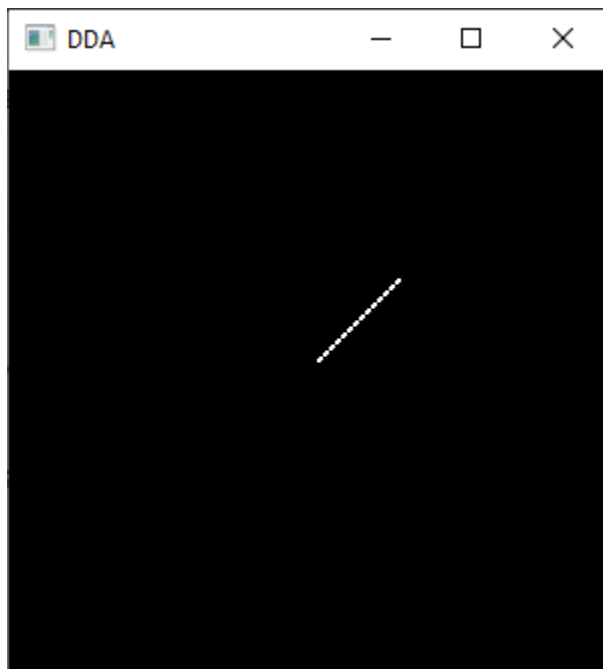
```
Enter x1 : 2
Enter y1 : 3
Enter x2 : 30
Enter y2 : 30
m=0.964286
(2.00,3.00)
(3.000000,3.964286)
(4.000000,4.928572)
(5.000000,5.892858)
(6.000000,6.857143)
(7.000000,7.821429)
(8.000000,8.785715)
(9.000000,9.750001)
(10.000000,10.714287)
(11.000000,11.678573)
(12.000000,12.642859)
(13.000000,13.607144)
(14.000000,14.571430)
(15.000000,15.535716)
(16.000000,16.500002)
(17.000000,17.464287)
(18.000000,18.428572)
(19.000000,19.392857)
(20.000000,20.357141)
(21.000000,21.321426)
(22.000000,22.285711)
(23.000000,23.249996)
(24.000000,24.214281)
(25.000000,25.178566)
(26.000000,26.142851)
(27.000000,27.107136)
(28.000000,28.071421)
(29.000000,29.035706)
(30.000000,29.999990)
```

$m > 1$  :

```
C:\Vikram\Vikram_SEM-7\Graphic
Enter x1 : 3
Enter y1 : 4
Enter x2 : 30
Enter y2 : 40
m=1.333333
(3.00,4.00)
(3.750000,5.000000)
(4.500000,6.000000)
(5.250000,7.000000)
(6.000000,8.000000)
(6.750000,9.000000)
(7.500000,10.000000)
(8.250000,11.000000)
(9.000000,12.000000)
(9.750000,13.000000)
(10.500000,14.000000)
(11.250000,15.000000)
(12.000000,16.000000)
(12.750000,17.000000)
(13.500000,18.000000)
(14.250000,19.000000)
(15.000000,20.000000)
(15.750000,21.000000)
(16.500000,22.000000)
(17.250000,23.000000)
(18.000000,24.000000)
(18.750000,25.000000)
(19.500000,26.000000)
(20.250000,27.000000)
(21.000000,28.000000)
(21.750000,29.000000)
(22.500000,30.000000)
(23.250000,31.000000)
(24.000000,32.000000)
(24.750000,33.000000)
(25.500000,34.000000)
(26.250000,35.000000)
(27.000000,36.000000)
(27.750000,37.000000)
(28.500000,38.000000)
(29.250000,39.000000)
(30.000000,40.000000)
```



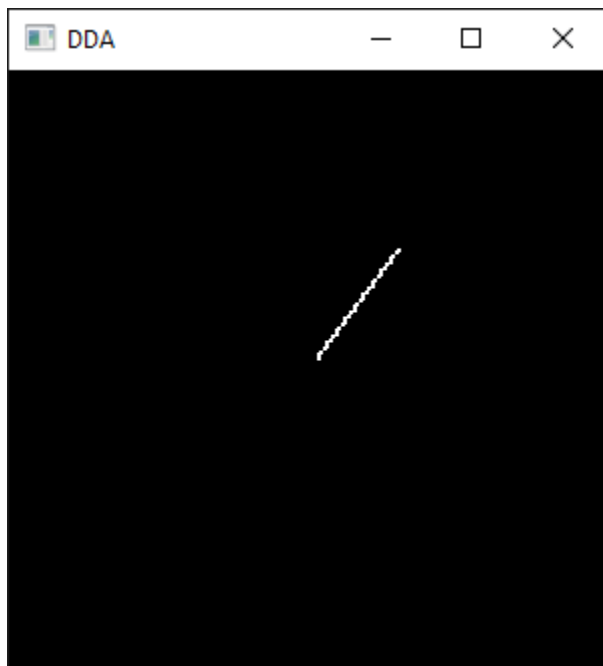
**+ve slope right to left :**  
 **$m \leq 1$ :**



C:\Vikram\Vikram\_SEM-7\Graphi

```
Enter x1 : 30
Enter y1 : 30
Enter x2 : 3
Enter y2 : 4
m=0.962963
(30.00,30.00)
(29.000000,29.037037)
(28.000000,28.074074)
(27.000000,27.111111)
(26.000000,26.148148)
(25.000000,25.185184)
(24.000000,24.222221)
(23.000000,23.259258)
(22.000000,22.296295)
(21.000000,21.333332)
(20.000000,20.370369)
(19.000000,19.407406)
(18.000000,18.444443)
(17.000000,17.481480)
(16.000000,16.518517)
(15.000000,15.555553)
(14.000000,14.592590)
(13.000000,13.629627)
(12.000000,12.666664)
(11.000000,11.703701)
(10.000000,10.740738)
(9.000000,9.777775)
(8.000000,8.814812)
(7.000000,7.851849)
(6.000000,6.888885)
(5.000000,5.925922)
(4.000000,4.962959)
(3.000000,3.999996)
```

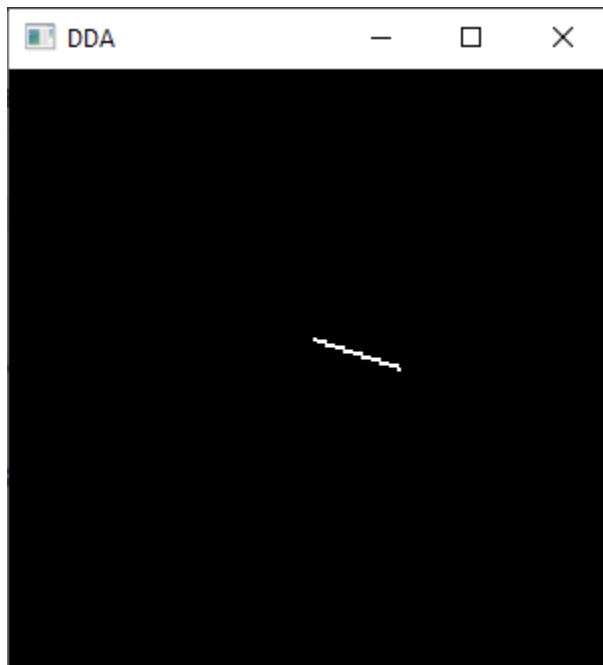
$m > 1$ :



C:\Vikram\Vikram\_SEM-7\Grap

```
Enter x1 : 30
Enter y1 : 40
Enter x2 : 3
Enter y2 : 4
m=1.333333
(30.00,40.00)
(29.250000,39.000000)
(28.500000,38.000000)
(27.750000,37.000000)
(27.000000,36.000000)
(26.250000,35.000000)
(25.500000,34.000000)
(24.750000,33.000000)
(24.000000,32.000000)
(23.250000,31.000000)
(22.500000,30.000000)
(21.750000,29.000000)
(21.000000,28.000000)
(20.250000,27.000000)
(19.500000,26.000000)
(18.750000,25.000000)
(18.000000,24.000000)
(17.250000,23.000000)
(16.500000,22.000000)
(15.750000,21.000000)
(15.000000,20.000000)
(14.250000,19.000000)
(13.500000,18.000000)
(12.750000,17.000000)
(12.000000,16.000000)
(11.250000,15.000000)
(10.500000,14.000000)
(9.750000,13.000000)
(9.000000,12.000000)
(8.250000,11.000000)
(7.500000,10.000000)
(6.750000,9.000000)
(6.000000,8.000000)
(5.250000,7.000000)
(4.500000,6.000000)
(3.750000,5.000000)
(3.000000,4.000000)
```

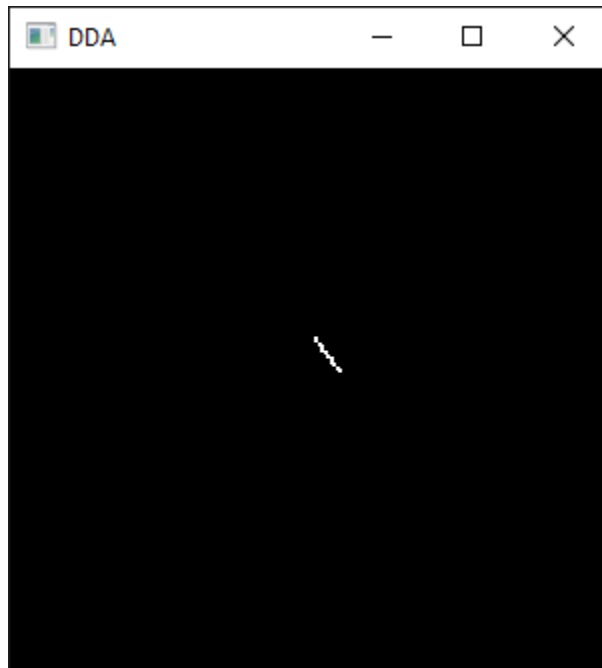
-ve slope left to right:  
 $|m| \leq 1$  :



```
C:\Vikram\Vikram_SEM-7\Grapl
Enter x1 : 2
Enter y1 : 10
Enter x2 : 30
Enter y2 : 1
m=-0.321429
(2.00,10.00)
(3.000000,9.678572)
(4.000000,9.357143)
(5.000000,9.035715)
(6.000000,8.714287)
(7.000000,8.392859)
(8.000000,8.071430)
(9.000000,7.750001)
(10.000000,7.428573)
(11.000000,7.107144)
(12.000000,6.785715)
(13.000000,6.464286)
(14.000000,6.142858)
(15.000000,5.821429)
(16.000000,5.500000)
(17.000000,5.178571)
(18.000000,4.857142)
(19.000000,4.535714)
(20.000000,4.214285)
(21.000000,3.892856)
(22.000000,3.571428)
(23.000000,3.249999)
(24.000000,2.928571)
(25.000000,2.607142)
(26.000000,2.285714)
(27.000000,1.964285)
(28.000000,1.642857)
(29.000000,1.321428)
(30.000000,1.000000)
```



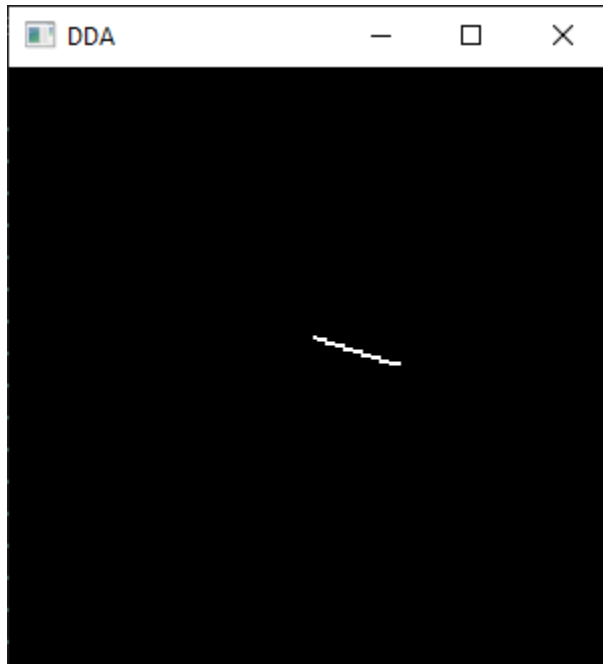
$|m| > 1$ :



C:\Vikram\Vikram\_SEM-7\Graphi

```
Enter x1 : 2
Enter y1 : 10
Enter x2 : 10
Enter y2 : -1
m=-1.375000
(2.00,10.00)
(2.727273,9.000000)
(3.454545,8.000000)
(4.181818,7.000000)
(4.909091,6.000000)
(5.636364,5.000000)
(6.363637,4.000000)
(7.090910,3.000000)
(7.818183,2.000000)
(8.545456,1.000000)
(9.272729,0.000000)
(10.000002,-1.000000)
```

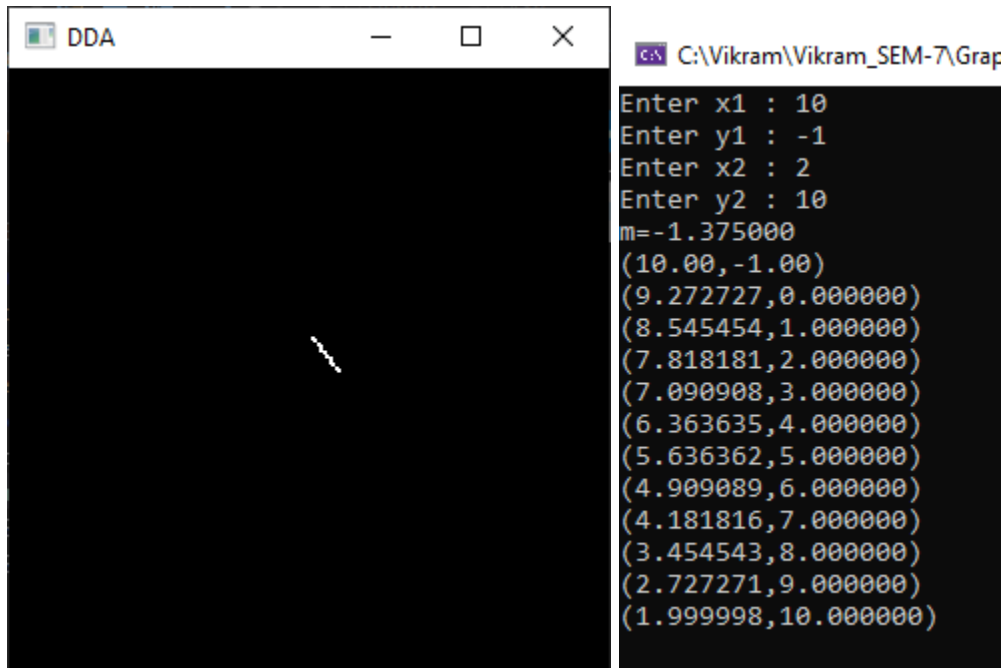
-ve slope right to left :  
 $|m| \leq 1$  :



C:\Vikram\Vikram\_SEM-7\Graph

```
Enter x1 : 30
Enter y1 : 1
Enter x2 : 2
Enter y2 : 10
m=-0.321429
(30.00,1.00)
(29.000000,1.321429)
(28.000000,1.642857)
(27.000000,1.964286)
(26.000000,2.285714)
(25.000000,2.607143)
(24.000000,2.928571)
(23.000000,3.250000)
(22.000000,3.571428)
(21.000000,3.892857)
(20.000000,4.214285)
(19.000000,4.535714)
(18.000000,4.857143)
(17.000000,5.178572)
(16.000000,5.500000)
(15.000000,5.821429)
(14.000000,6.142858)
(13.000000,6.464287)
(12.000000,6.785716)
(11.000000,7.107144)
(10.000000,7.428573)
(9.000000,7.750002)
(8.000000,8.071430)
(7.000000,8.392859)
(6.000000,8.714287)
(5.000000,9.035715)
(4.000000,9.357143)
(3.000000,9.678572)
(2.000000,10.000000)
```

$|m| > 1$  :



## **CONCLUSION :**

Thus the DDA line drawing algorithm was implemented and the points were plotted satisfying all the 8 test cases.