Date
Page

Assignment No - A-6

Student Name:- Sameer Manoj Bramhecha
Roll No:- 21115
Batch:- E-1
Date of Performance:- 28/09/2021
Date of Submission:- 05/10/2021

Title:- Pattern Drawing using line and circle.

Problem Statement:→ Write C++ program to draw a given pattern. Use DDA line and Bresenham's circle drawing algorithm. Apply the concept of encapsulation.

Learning Objectives:- To understand and implement DDA line and Bresenham's Circle drawing algorithm.

S/W and H/W requirements:-
1.) Basic programming skills of C++.
2.) 64-bit Open Source linux.
3.) Open Source C++ programming tool like G++/GCC.

Theory:-
DDA Line Drawing Algorithm:-
Line is a basic element in graphics. To draw a line you need two end points between which you can draw a line. ~~DE~~

Date
Page

Digital Differential Analyzer (DDA) is a simple, incremental line scan converting algorithm.
In the DDA algorithm, either horizontal or vertical displacement is set to unit interval and the corresponding ~~displa~~ displacement for other direction is calculated using the slope.
If the line makes an angle less than 45° with X-axis (i.e $m<1$), increment in the x-direction is set to 1 and corresponding y is computed.
If the line makes an angle greater than 45° with X-axis (i.e $m>1$), increment in the Y-direction is set to 1 and corresponding x is computed.
For line with slope $m=1$, increment in both the directions is set to 1.

Advantages of DDA line drawing algorithm:-

1. It is simple, easy to understand and quite faster.
2. It eliminates multiplications involved in explicit line drawing equation, $y=mx+c$.
3. It is more efficient than an implicit line drawing algorithm.

Disadvantages of DDA line drawing algorithm:-
1. It involves floating-point operation for each pixel.
2. It performs rounding off operation for each pixel.
3. Rounding off error is accumulated in each iteration and calculated pixel position may drift away from the actual position due to cumulative rounding off error.

Date

Page

* Bresenham's Circle Drawing Algorithm:→

This is an incremental algorithm. The circle is defined by the equation. $x^2+y^2=r^2$. The algorithm starts drawing the circle from the initial point and computes E(epsilon), by which the x and y coordinates should be increased or decreased. The increment E is derived as follow~~s~~

find the n such that,

$$2^{n-1} \le r \le 2^n$$

Where r is the radius of circle

$$\varepsilon = \frac{1}{2^n}$$

Start drawing the circle from point (0,r), using E, we can compute the points along circle periphery as follow:

$$x_2 = x_1 + \varepsilon y_1$$

$$y_2 = y_1 - \varepsilon x_1$$

Computation halts when $((y_1-y_0)<\varepsilon$ or $(x_1-x_0)>\varepsilon)$

Advantages of Bresenham's Circle Drawing Algorithm:-

① Derivation of decision parameters is based on simple circle drawing formula, i.e $x^2+y^2=r^2$

② It is easy to implement.

Disadvantages:-

① It is time-consuming.

② We won't get smooth circle due to uneven distance between pixels.

Date
Page

Algorithm:-

① DDA Line drawing algorithm:->

```
1.> Start
2.> Read endpoints $(x_1, y_1)$ and $(x_2, y_2)$ from user.
3> Calculate $dx = x_2 - x_1$ and $dy = y_2 - y_1$
4.) if ( abs (dx) >= abs (dy))
        step = abs (dx)
    else
        step = abs (dy)
5.> $x = x_1$, $y = y_1$, $i = 0$
6.> $xinc = \frac{dx}{step}$, $yinc = \frac{dy}{step}$.
7.) while (i <= step)
    {
        setpixel (x, y, color)
        x = x + xinc;
        y = y + yinc;
        i ++
    }
8.) end.
```

② Bresenham's Circle Drawing Algorithm:-

```
1.> Start.
2.> Read centre (xc, yc) and radius r for circle.
3.) set x = 0 and y = r.
4) D = 3 - (2 * r)   // Initial Decision parameter.
5.) while (x <= y)
    {   putpixel (y+xc, x+yc; 15);     // octet 1
        putpixel (x+xc, y+yc; 15);     // octet 2
        putpixel (-x+xc, y+yc, 15);    // octet 3
        putpixel (-y+xc, x+xc, 15);    // octet 4.
```

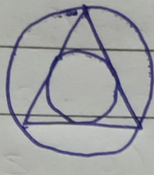
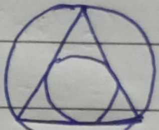
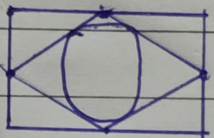
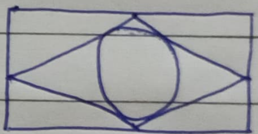
```
putpixel(-y+xc, -x+yc, 15);   // Octet -~~4~~5
putpixel(-x+xc, -y+yc, 15);   // octet -~~5~~6
putpixel(x+xc, -y+yc, 15);    // octet -~~6~~7
putpixel(y+xc, -x+yc, 15);    // octet - 8

if (D<0)
{  D = D + 4*x + 6;  //Next Decision Parameter
}
else {
D = D + 4*(x-y) + 10;  //Next Decision Parameter
   y = y-1; }
x = x+1
}
```

6.) Stop.

Date

Page

Test cases:-

Test case No.	Test Case Description	Input	Expected Output	Actual Output	Result
1.	Pattern 1.	$x_c = 200$ $y_c = 200$ $r = 100$.			Pass.
2.	Pattern:	length = 300 breadth = 100 $x_1 = 200$ $y_1 = 200$			Pass.

Conclusion:-

We have successfully implemented given patterns using DDA line and Bresenham's circle algorithm.