****

**UCS1712-Graphics and Multimedia Lab**

**Programming Assignment 1**

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

Name: Jayannthan P T

Dept: CSE ‘A’

Roll No.: 205001049

a) To plot points that make up the line with endpoints (x0,y0) and (xn,yn) using 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|<= 1 (ii) |m|>1

Note that all four cases of line drawing must be given as test cases.

**Source code:**

include <iostream>

#include <cmath>

#include <GLUT/glut.h>

using namespace std;

int choice = 0, flag = 0;

float x\_1 = 0, y\_1 = 0, x\_2 = 0, y\_2 = 0;

string cnt = "YES";

void **myInit**()

{

**glClearColor**(1.0, 1.0, 1.0, 0.0);

**glPointSize**(2);

**glMatrixMode**(GL\_PROJECTION);

**glLoadIdentity**();

**gluOrtho2D**(0.0, 640.0, 0.0, 480.0);

}

void **drawLine**()

{

**glBegin**(GL\_POINTS); // Begin drawing points x\_1 += 320;

    x\_2 += 320;

    y\_1 += 240;

    y\_2 += 240;

    float dx = x\_2 - x\_1;

    float dy = y\_2 - y\_1;

    int steps;

    if (**abs**(dx) > **abs**(dy))

        steps = **abs**(dx);

    else

        steps = **abs**(dy);

    float x = x\_1, y = y\_1; // Start from the actual (x\_1, y\_1) point

    float m = dy / dx;

    for (int i = 0; i <= steps; i++)

    {

**glVertex2f**(**round**(x), **round**(y)); // Draw the current point

        if (choice == 1)

        {

            x += 1;

            y += m;

        }

        else if (choice == 2)

        {

            x -= 1;

            y -= m;

        }

        else if (choice == 3)

        {

            x += 1;

            y -= m;

        }

        else

        {

            x -= 1;

            y += m;

        }

    }

**glEnd**(); // End drawing points

}

void **myDisplay**()

{

**glClear**(GL\_COLOR\_BUFFER\_BIT);

**glColor3f**(1.0f, 0.0f, 0.0f);

    if (flag == 0)

    {

        flag = 1;

        x\_1 = 0, y\_1 = 0, x\_2 = 0, y\_2 = 0;

        cout **<<** "Point 1 : ";

        cin **>>** x\_1 **>>** y\_1;

        cout **<<** "Point 2 : ";

        cin **>>** x\_2 **>>** y\_2;

        int m = (y\_2 - y\_1) / (x\_2 - x\_1);

        if (x\_1 < x\_2 && **abs**(m) <= 1)

        {

            choice = 1;

        }

        else if (x\_2 > x\_1 && **abs**(m) <= 1)

        {

            choice = 2;

        }

        else if (x\_1 < x\_2 && **abs**(m) > 1)

        {

            choice = 3;

        }

        else

        {

            choice = 4;

        }

    }

**drawLine**();

**glFlush**();

    cout **<<** "Want to continue (YES/NO) : ";

    cin **>>** cnt;

    if (cnt **==** "NO")

    {

        cout **<<** "Exiting...\n";

**exit**(0);

    }

    flag = 0;            // Reset flag for the next line

**glutPostRedisplay**(); // Continue updating the display

}

int **main**(int argc, char \*\*argv)

{

**glutInit**(&argc, argv);

**glutInitDisplayMode**(GLUT\_SINGLE | GLUT\_RGB);

**glutInitWindowSize**(640, 480);

**glutCreateWindow**("Line Drawing Example");

**myInit**();

**glutDisplayFunc**(**myDisplay**);

**glutMainLoop**();

    return 0;

}









