#include <GL/glut.h>

#include <iostream>

using namespace std;

int lineType = 0;

bool triangleDrawn = false;

// Initialize OpenGL

void init() {

glClearColor(1, 1, 1, 1); // White background

gluOrtho2D(0, 800, 0, 800); // 2D projection

glPointSize(2); // Pixel size

}

// Draw a single pixel

void drawPixel(float x, float y) {

glBegin(GL\_POINTS);

glVertex2f(x, y);

glEnd();

}

// DDA Algorithm with Line Styles

void drawLineDDA(int x1, int y1, int x2, int y2, int type) {

float dx = x2 - x1;

float dy = y2 - y1;

int steps = max(abs(dx), abs(dy));

float xInc = dx / steps;

float yInc = dy / steps;

float x = x1, y = y1;

for (int i = 0; i <= steps; i++) {

bool draw = false;

if (type == 0) draw = true; // Simple

else if (type == 1 && i % 5 == 0) draw = true; // Dotted

else if (type == 2 && (i % 10 < 5)) draw = true; // Dashed

else if (type == 3) { // Solid line

glLineWidth(4);

glBegin(GL\_LINES);

glVertex2f(x1, y1);

glVertex2f(x2, y2);

glEnd();

return;

}

if (draw)

drawPixel(x, y);

x += xInc;

y += yInc;

}

}

// Draw Axis

void drawAxis() {

glColor3f(0, 0, 0);

drawLineDDA(0, 400, 800, 400, 0);

drawLineDDA(400, 0, 400, 800, 0);

}

// Draw Triangle

void drawTriangle() {

glColor3f(0, 0, 1); // Blue triangle

drawLineDDA(400, 500, 500, 300, lineType);

drawLineDDA(500, 300, 300, 300, lineType);

drawLineDDA(300, 300, 400, 500, lineType);

}

// Display callback

void display() {

glClear(GL\_COLOR\_BUFFER\_BIT);

drawAxis();

if (triangleDrawn)

drawTriangle();

glFlush();

}

// Menu callback

void menu(int option) {

lineType = option;

glutPostRedisplay();

}

// Mouse callback

void mouse(int button, int state, int x, int y) {

if (button == GLUT\_LEFT\_BUTTON && state == GLUT\_DOWN) {

triangleDrawn = true;

glutPostRedisplay();

}

}

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

glutInit(&argc, argv);

glutInitWindowSize(800, 800);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutCreateWindow("DDA Triangle Drawing");

init();

glutDisplayFunc(display);

glutMouseFunc(mouse);

// Right-click menu

glutCreateMenu(menu);

glutAddMenuEntry("Simple Line", 0);

glutAddMenuEntry("Dotted Line", 1);

glutAddMenuEntry("Dashed Line", 2);

glutAddMenuEntry("Solid Line", 3);

glutAttachMenu(GLUT\_RIGHT\_BUTTON);

glutMainLoop();

return 0;

}