**UCS1712 – GRAPHICS AND MULTIMEDIA LAB**

**Lab Exercise 7: Cohen Sutherland Line clipping in C++ using OpenGL**

CODE:

#include<gl/glut.h>

#include<iostream>

#include<utility>

using namespace std;

pair<int, int> P1, P2;

int X1, X2, Y1, Y2;

int xwmin, xwmax, ywmin, ywmax;

void myInit()

{

glClearColor(1.0, 1.0, 1.0, 0.0);

glColor3f(0.0f, 0.0f, 0.0f);

glPointSize(10);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(0.0, 640.0, 0.0, 480.0);

}

void drawWindow() {

glBegin(GL\_LINE\_LOOP);

glVertex2d(xwmin, ywmin);

glVertex2d(xwmax, ywmin);

glVertex2d(xwmax, ywmax);

glVertex2d(xwmin, ywmax);

glEnd();

}

void drawOriginal() {

glBegin(GL\_LINES);

glVertex2d(P1.first, P1.second);

glVertex2d(P2.first, P2.second);

glEnd();

}

int getRC(pair<int, int>& P)

{

int rc = 0;

if (P.first < xwmin) rc |= 1;

else if (P.first > xwmax) rc |= 1 << 1;

if (P.second < ywmin) rc |= 1 << 2;

else if (P.second > ywmax) rc |= 1 << 3;

return rc;

}

void findIntersection(pair<int, int>& P, double m, int rc) {

if (rc == 0) return;

// y = ywmax

if ((rc >> 3)&1) {

//x =X1 + (y-Y1)/m

P.second = ywmax;

P.first = X1 + (ywmax - Y1) / m;

return;

}

//y = ywmin

if ((rc >> 2 )& 1) {

//x =X1 + (y-Y1)/m

P.second = ywmin;

P.first = X1 + (ywmin - Y1) / m;

return;

}

// x= xwmax

if ((rc >> 1) & 1) {

//y =Y1 + (x-X1)\*m

P.first = xwmax;

P.second = Y1 + (xwmax - X1) \* m;

return;

}

// x= xwmin

if (rc & 1) {

//y =Y1 + (x-X1)\*m

P.first = xwmin;

P.second = Y1 + (xwmin - X1) \* m;

return;

}

}

void PerformClipping(pair<int, int>& P1, pair<int, int>& P2)

{

int rc1 = getRC(P1), rc2 = getRC(P2);

//Checking for trivial OR

if (int(rc1 | rc2) == 0) {

glBegin(GL\_LINES);

glVertex2d(P1.first, P1.second);

glVertex2d(P2.first, P2.second);

glEnd();

return;

}

else if (int(rc1 & rc2) != 0) return;

double m = (Y2-Y1) \* 1.0 / (X2-X1);

findIntersection(P1, m, rc1);

findIntersection(P2, m, rc2);

PerformClipping(P1, P2);

}

void myDisplay()

{

glClear(GL\_COLOR\_BUFFER\_BIT);

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

drawWindow();

glColor3f(0.0f, 0.0f, 0.0f);

drawOriginal();

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

PerformClipping(P1, P2);

glFlush();

}

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

{

cout << "Enter window properties:" << endl;

cout << "xwmin:";

cin >> xwmin;

cout << "xwmax:";

cin >> xwmax;

cout << "ywmin:";

cin >> ywmin;

cout << "ywmax:";

cin >> ywmax;

int x, y;

cout << endl << "Enter point p1(x,y) :";

cin >> x >> y;

P1.first = x;

P1.second = y;

X1 = x;

Y1 = y;

cout << "Enter point p2(x,y) :";

cin >> x >> y;

P2.first = x;

P2.second = y;

X2 = x;

Y2 = y;

cout << "Blue -> Clipping Window" << endl;

cout << "Black -> Original Line" << endl;

cout << "Red -> Clipped Line" << endl;

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize(640, 480);

glutCreateWindow("Cohen Sutherland");

glutDisplayFunc(myDisplay);

myInit();

glutMainLoop();

return 1;

}

OUTPUT:



