#include<iostream>

#include<graphics.h>

#include<windows.h>

using namespace std;

static int x\_max, y\_max, x\_min, y\_min;

static int INSIDE = 0;

static int LEFT = 1;

static int RIGHT = 2;

static int BOTTOM = 4;

static int TOP = 8;

int binarycode(double x, double y) {

int code = INSIDE;

if (x < x\_min)

code |= LEFT;

else if (x > x\_max)

code |= RIGHT;

if (y < y\_min)

code |= BOTTOM;

else if (y > y\_max)

code |= TOP;

return code;

}

void cohensutherlandclip(double x1, double y1, double x2, double y2) {

int code1 = binarycode(x1, y1);

int code2 = binarycode(x2, y2);

bool accept = false;

while (true) {

if ((code1 == 0) && (code2 == 0)) {

accept = true;

break;

} else if (code1 & code2) {

break;

} else {

int code\_out;

double x, y;

if (code1 != 0)

code\_out = code1;

else

code\_out = code2;

if (code\_out & TOP) {

x = x1 + (x2 - x1) \* (y\_max - y1) / (y2 - y1);

y = y\_max;

} else if (code\_out & BOTTOM) {

x = x1 + (x2 - x1) \* (y\_min - y1) / (y2 - y1);

y = y\_min;

} else if (code\_out & RIGHT) {

y = y1 + (y2 - y1) \* (x\_max - x1) / (x2 - x1);

x = x\_max;

} else if (code\_out & LEFT) {

y = y1 + (y2 - y1) \* (x\_min - x1) / (x2 - x1);

x = x\_min;

}

if (code\_out == code1) {

x1 = x;

y1 = y;

code1 = binarycode(x1, y1);

} else {

x2 = x;

y2 = y;

code2 = binarycode(x2, y2);

}

}

}

if (accept) {

Sleep(100); // use delay or sleep

rectangle(x\_min, y\_min, x\_max, y\_max);

line(x1, y1, x2, y2);

}

}

int main() {

int gdriver = DETECT, gmode;

initgraph(&gdriver, &gmode, NULL);

cout << "Enter left, top, and right, bottom of the window: ";

cin >> x\_min >> y\_min >> x\_max >> y\_max;

rectangle(x\_min, y\_min, x\_max, y\_max);

int x1, y1, x2, y2;

cout << "Enter the endpoints of the line: ";

cin >> x1 >> y1 >> x2 >> y2;

Sleep(500); // use delay or sleep

cleardevice();

cohensutherlandclip(x1, y1, x2, y2);

getch();

closegraph();

return 0;

}