#include <iostream>

#include <graphics.h> //graphics.h library is used to include graphical operations in a program.

#include <math.h>

using namespace std;

class scan

{

public:

int x[20], y[20], k;

float slope[20], x\_int[20];

void polygon(int n);

};

void scan::polygon(int n)

{

int i;

float dx, dy;

x[n] = x[0];

y[n] = y[0];

for (int i = 0; i < n; i++) // draw all lines (edges of polygon)

{

line(x[i], y[i], x[i + 1], y[i + 1]); // line cordinates x1,y1,x2,y2

}

for (i = 0; i < n; i++) // finding slope of all lines

{

dy = y[i + 1] - y[i]; // dy=y2-y1

dx = x[i + 1] - x[i]; // dx=x2-x1

// if(dy==0)

// slope[i]=1;

// else if(dx==0)

// slope[i]=0;

// else

slope[i] = dx / dy;

}

// finding intersection points

for (int p = 0; p < 480; p++) // consider 480 horizontal lines on screen

{

k = 0;

for (i = 0; i < n; i++)

{

if (((y[i] <= p) && (y[i + 1] > p)) || ((y[i] > p) && (y[i + 1] <= p)))

{

x\_int[k] = x[i] + slope[i] \* (p - y[i]); // find out intersection points using formula

k++;

}

}

for (int j = 0; j < k - 1; j++) // perform sorting of intersection points on x direction

{

for (int i = 0; i < k - 1; i++)

{

if (x\_int[i] > x\_int[i + 1])

{

int temp = x\_int[i];

x\_int[i] = x\_int[i + 1];

x\_int[i + 1] = temp;

}

}

}

for (int i = 0; i < k; i = i + 2) // fill points of line that are interior to polygon

{

setcolor(YELLOW);

line(x\_int[i], p, x\_int[i + 1], p); // x1,y1,x2,y2

delay(50);

}

}

}

int main()

{

int n, i;

scan p;

cout << "Enter edge : \t";

cin >> n;

cout << "\n\nEnter Coordinates : \t";

for (i = 0; i < n; i++)

{

cin >> p.x[i] >> p.y[i];

}

int gd, gm;

gd = DETECT;

initgraph(&gd, &gm, NULL);

p.polygon(n);

getch();

closegraph();

return 0;

}