#define BLACK 0

#include <stdlib.h>

#include <stdio.h>

#include <GL/glut.h>

int submenu;

float x1,x2,x3,x4,y1,y2,y3,y4;

void edgedetect(float x1,float y1,float x2,float y2,int \*le,int \*re)

{

float mx,x,temp;

int i;

if((y2-y1)<0)

{

temp=y1;y1=y2;y2=temp;

temp=x1;x1=x2;x2=temp;

}

if((y2-y1)!=0)

mx=(x2-x1)/(y2-y1);

else

mx=x2-x1;

x=x1;

for(i=y1;i<=y2;i++)

{

if(x<(float)le[i])

le[i]=(int)x;

if(x>(float)re[i])

re[i]=(int)x;

x+=mx;

}

}

void draw\_pixel(int x,int y,int value)

{

//glColor3f(1.0,1.0,0.0);

glBegin(GL\_POINTS);

glVertex2i(x,y);

glEnd();

}

void scanfill(float x1,float y1,float x2,float y2,float x3,float y3,float x4,float y4)

{

int le[500],re[500];

int i,y;

for(i=0;i<500;i++)

{

le[i]=500;

re[i]=0;

}

edgedetect(x1,y1,x2,y2,le,re);

edgedetect(x2,y2,x3,y3,le,re);

edgedetect(x3,y3,x4,y4,le,re);

edgedetect(x4,y4,x1,y1,le,re);

for(y=0;y<500;y++)

{

if(le[y]<=re[y])

for(i=(int)le[y];i<(int)re[y];i++)

draw\_pixel(i,y,BLACK);

}

}

void display()

{

x1=200.0;y1=200.0;x2=100.0;y2=300.0;x3=200.0;y3=400.0;x4=300.0;y4=300.0;

glClear(GL\_COLOR\_BUFFER\_BIT);

glBegin(GL\_LINE\_LOOP);

glVertex2f(x1,y1);

glVertex2f(x2,y2);

glVertex2f(x3,y3);

glVertex2f(x4,y4);

glEnd();

scanfill(x1,y1,x2,y2,x3,y3,x4,y4);

glFlush();

}

void myinit()

{

glClearColor(1.0,1.0,1.0,1.0);

//glColor3f(1.0,0.0,0.0);

glPointSize(1.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(0.0,499.0,0.0,499.0);

}

void menufunc(int n)

{

switch(n)

{

case 1: glColor3f(1.0, 0.0, 0.0);

break;

case 2:

glColor3f(0.0, 1.0, 0.0);

break;

case 3: glColor3f(0.0, 0.0, 1.0);

}

}

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

{

glutInit(&argc,argv);

glutInitDisplayMode(GLUT\_SINGLE|GLUT\_RGB);

glutInitWindowSize(500,500);

glutInitWindowPosition(0,0);

glutCreateWindow("Filling a Polygon using Scan-line Algorithm");

submenu=glutCreateMenu(menufunc);

glutCreateMenu(menufunc);

glutAddMenuEntry("RED",1);

glutAddMenuEntry("GREEN",2);

glutAddMenuEntry("BLUE",3);

glutAttachMenu(GLUT\_RIGHT\_BUTTON);

glutDisplayFunc(display);

myinit();

glutMainLoop();

return 0;

}