|  |
| --- |
| #include<stdio.h> |
|  | #include<GL/glut.h> |
|  |  |
|  | GLfloat x1,y1,x2,y2,x3,y3,x4,y4,x5,y5; |
|  |  |
|  | void edgedetect(GLfloat x1,GLfloat y1,GLfloat x2,GLfloat y2,int \*le1,int \*re1,int \*le2,int \*re2) |
|  | { |
|  | float m,x,mx,temp; |
|  | if(y1>y2){ |
|  | 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(int i=y1;i<y2;i++){ |
|  | if(le2[i]<2 && re2[i]<2){ |
|  | if(x<le1[i]) |
|  | { |
|  | le1[i]=x; |
|  | le2[i]++; |
|  | if(le2[i]==2){ |
|  | le2[i]=500; |
|  | re2[i]=0; |
|  | } |
|  |  |
|  | } |
|  | if(x>re1[i]){ |
|  | re1[i]=x; |
|  | re2[i]++; |
|  | if(re2[1]==2){ |
|  | le2[i]=500; |
|  | re2[i]=0; |
|  |  |
|  | } |
|  | } |
|  | } |
|  | else{ |
|  | if(x<le2[i]){ |
|  | le2[i]=x; |
|  | } |
|  | if(x>re2[i]){ |
|  | re2[i]=x; |
|  | } |
|  | } |
|  | x=x+mx; |
|  | } |
|  | } |
|  | void drawpixel(int x,int y) |
|  | { |
|  | glColor3f(0.0,0.0,1.0); |
|  | 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,float x5,float y5) |
|  | { |
|  |  |
|  | int le1[500],le2[500],re1[500],re2[500]; |
|  | for(int i=0;i<500;i++){ |
|  | le1[i]=500; |
|  | le2[i]=0; |
|  | re1[i]=0; |
|  | re2[i]=0; |
|  | } |
|  | edgedetect(x1,y1,x2,y2,le1,re1,le2,re2); |
|  | edgedetect(x2,y2,x3,y3,le1,re1,le2,re2); |
|  | edgedetect(x3,y3,x4,y4,le1,re1,le2,re2); |
|  | edgedetect(x4,y4,x5,y5,le1,re1,le2,re2); |
|  | edgedetect(x5,y5,x1,y1,le1,re1,le2,re2); |
|  | for(int i=0;i<500;i++){ |
|  | if(le1[i]<=re1[i]){ |
|  | for(int j=le1[i]+1;j<re1[i];j++) |
|  | { |
|  | drawpixel(j,i); |
|  | } |
|  | } |
|  | if(le2[i]>2 && re2[i]>2){ |
|  | for(int j=le2[i]+1;j<re2[i];j++) |
|  | { |
|  | drawpixel(j,i); |
|  | } |
|  | } |
|  | } |
|  | } |
|  | void display() |
|  | { |
|  | glClear(GL\_COLOR\_BUFFER\_BIT); |
|  | //x1=200,y1=200,x2=100,y2=300,x3=200,y3=400,x4=300,y4=300; |
|  | x1=100,y1=100,x2=400,y2=100,x3=400,y3=400,x4=250,y4=300,x5=100,y5=400; |
|  | glBegin(GL\_LINE\_LOOP); |
|  | glVertex2f(x1,y1); |
|  | glVertex2f(x2,y2); |
|  | glVertex2f(x3,y3); |
|  | glVertex2f(x4,y4); |
|  | glVertex2f(x5,y5); |
|  | glEnd(); |
|  | scanfill(x1,y1,x2,y2,x3,y3,x4,y4,x5,y5); |
|  | glFlush(); |
|  | } |
|  | void myinit() |
|  | { |
|  | glClearColor(1.0,1.0,1.0,1.0); |
|  | glColor3f(0.0,0.0,1.0); |
|  |  |
|  | gluOrtho2D(0.0,499.0,0.0,499.0); |
|  | } |
|  | int main(int argc,char \*\*argv) |
|  | { glutInit(&argc,argv); |
|  | glutInitDisplayMode(GLUT\_SINGLE|GLUT\_RGB); |
|  | glutInitWindowSize(500,500); |
|  | glutInitWindowPosition(0,0); |
|  | glutCreateWindow("scanfill"); |
|  | glutDisplayFunc(display); |
|  | myinit(); |
|  | glutMainLoop(); |
|  | return 0; |
|  | } |