**CX-5组数学创新实验第一次报告（分形）**

**队员：漆宇豪 郭志猛 陈天翔 邵戈文**

**1.实验目的**：

利用matlab中循环、判断、矩阵运算等操作实现点坐标的自动生成，并绘制出相应的图像。

**2.实验思想：**

前两个实验，我们采用循环的方式，找到通项公式，将点一步一步的表示出来并插入矩阵中.我们规定循环五次，即第一个for循环的1:5。之后我们利用寻找到的规律将，通过矩阵乘法旋转将后续的点找出来。

第三个实验大大增加了编程难度，也增加了自由度使结果多样化：我们使图形的内凸外凸、凸法、初始的形状以及分形次数可以选择。并且可以覆盖前两个实验的分型操作。

四种功能：  
1.内外凸：封闭多边形的凸出朝向内还是外；  
2.凸法：凸起的形状（可选四种）；  
3.最初图形形状：正n边形，可指定n的值；  
4.分形次数：可在性能允许范围内输入正整数的值。

第四个，我们自主创新设计新的分形，以三角形为基础生成五角星的方式设计分形。

第五个，我们设计出了GUI，直接在界面操作实现分形。

**3.实验结果：**

1. **旋转60度**

r=[cos(pi/3) -sin(pi/3);sin(pi/3) cos(pi/3)];

p=[0 1;0 0];

for n=1:5;

q=p;

p=zeros(2,4^n+1);%一共有4^n+1个点

for ii=1:4^(n-1);

x=q(:,ii+1)-q(:,ii);%坐标差

p(:,ii.\*4-3)=q(:,ii);%原来的第ii个点变为后来的第4\*ii-3个点

p(:,ii.\*4-2)=q(:,ii)+1/3\*x;%原来的第ii个点变为后来的第4\*ii-2个点

p(:,ii.\*4-1)=q(:,ii)+1/3\*x+1/3\*r\*x;%原来的第ii个点变为后来的第4\*ii-1个点

p(:,ii.\*4)=q(:,ii)+2/3\*x;%原来的第ii个点变为后来的第4\*ii个点

end

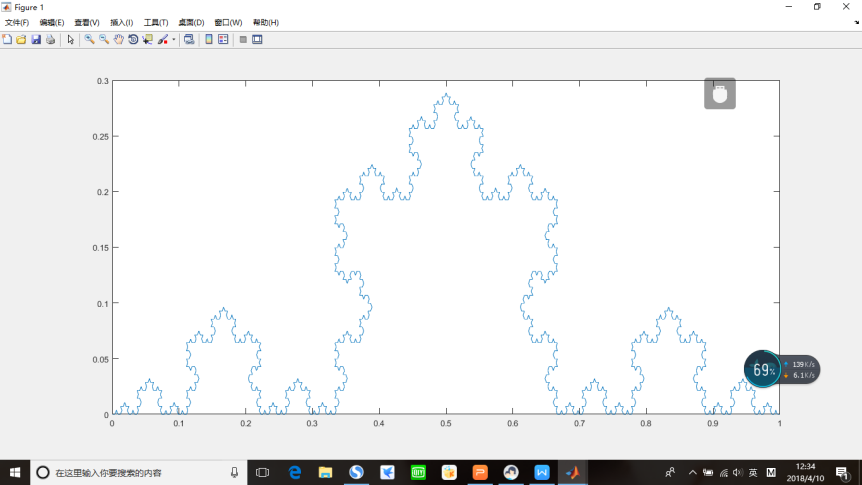
p(:,4^n+1)=[1 0]';%最后一个点坐标不变

end

x1=p(1,:);

y1=p(2,:);

plot(x1,y1)



1. **旋转90度**

r=[cos(pi/2) -sin(pi/2);sin(pi/2) cos(pi/2)];

p=[0 1;0 0];

for n=1:5;

q=p;

p=zeros(2,5^n+1);%一共有5^n+1个点

for ii=1:5^(n-1);

x=q(:,ii+1)-q(:,ii);%坐标差

p(:,ii.\*5-4)=q(:,ii);

p(:,ii.\*5-3)=q(:,ii)+1/3\*x;

p(:,ii.\*5-2)=q(:,ii)+1/3\*x+1/3\*r\*x;

p(:,ii.\*5-1)=q(:,ii)+2/3\*x+1/3\*r\*x;

p(:,ii.\*5)=q(:,ii)+2/3\*x;

end

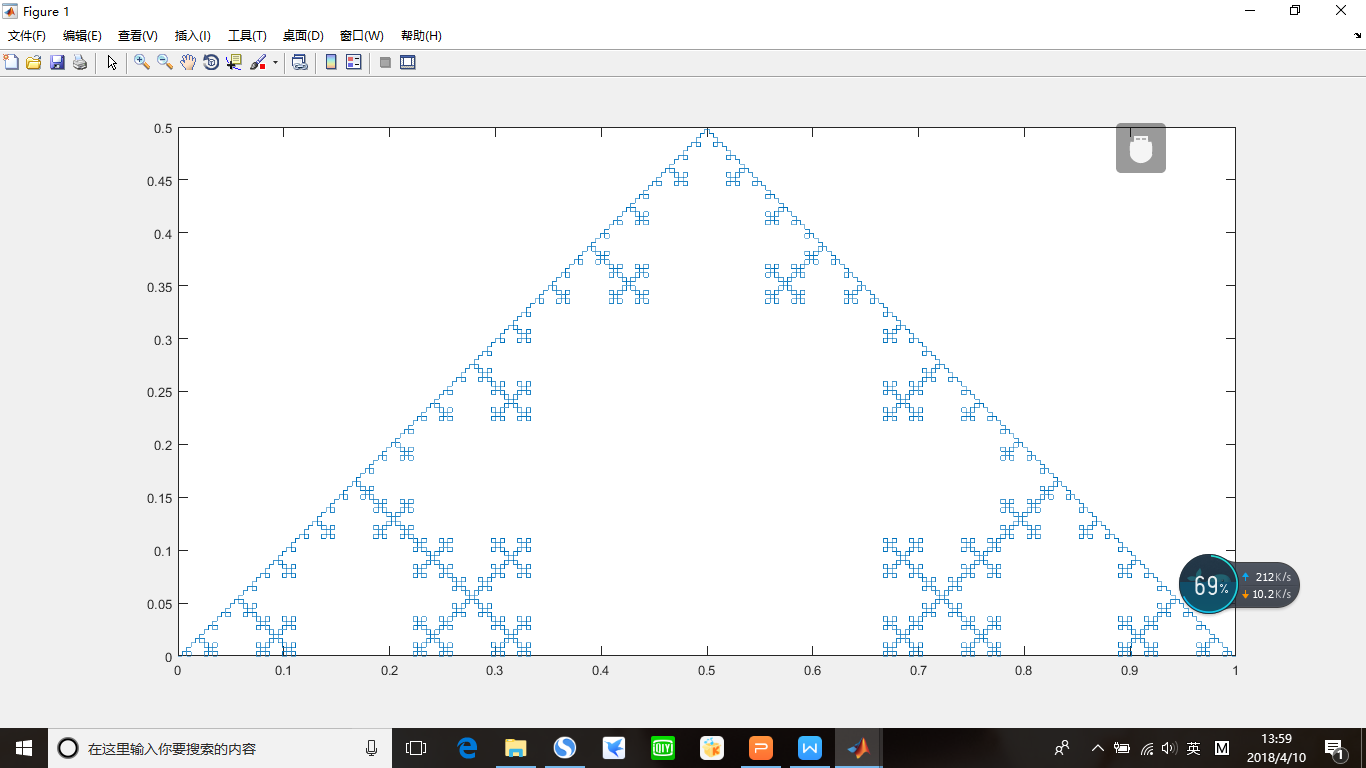
p(:,5^n+1)=[1 0]';%最后一个点坐标不变

end

x1=p(1,:);

y1=p(2,:);

plot(x1,y1)



**3.多功能绘图**

四种功能：

1.内外凸：封闭多边形的凸出朝向内还是外；

2.凸法：凸起的形状（可选四种）；

3.最初图形形状：正n边形，可指定n的值；

4.分形次数：可在性能允许范围内输入正整数的值。

clear

ti=4;%ti是用于分形时候隔ti个点分一次

in=input('你想让他向内还是向外凸（1表示内，-1表示外）：');

pp=[cos(in\*pi/3),-sin(in\*pi/3);sin(in\*pi/3) cos(in\*pi/3)];%默认的逆时针旋转pi/3

tufa=input('你要怎么个凸法（正三角形输1，直角三角形输2,正方形输3,树枝输4）：');

if tufa==2 %用这三个if是因为不同凸法可能会有不同的旋转方式和隔的点数

pp=[cos(in\*pi/2),-sin(in\*pi/2);sin(in\*pi/2) cos(in\*pi/2)];

end

if tufa==3

pp=[cos(in\*pi/2),-sin(in\*pi/2);sin(in\*pi/2) cos(in\*pi/2)];

ti=5;

end

if tufa==4

ti=7;

end

shape=input('你就说你要正几边形（例如3）：');%产生最开始的形状

p=[0 1 0;0 0 0];%初始的坐标

for k=3:shape %用于产生最开始的正shape边形

angle=pi\*2/shape\*(k-2);

q1=p(:,k-1)+[cos(angle);sin(angle)];

p=[p(:,1:k-1),q1,p(:,k:end)];

end

cishu=input('请输入分的次数（例如3）：'); %用于限定分形次数

for j=1:cishu %这个for进行一次是将形状分一次

n=size(p,2);

for i=1:ti:ti\*(n-1) %这个for在每两个点间分一次，全部进行完则整个形状分一次

p1=p(:,i);

p2=p(:,i+1);

p3=p1+(p2-p1)/3;

p4=2\*p3-p1;

p5=p3+pp\*(p4-p3);

if tufa==3 %这里if是因为不同凸法插入点个数不同

p6=p5+p4-p3;

p=[p(:,1:i),p3,p5,p6,p4,p(:,i+1:end)];

elseif tufa==4

p6=p4-pp\*(p4-p3);

p=[p(:,1:i),p3,p5,p3,p4,p6,p4,p(:,i+1:end)];

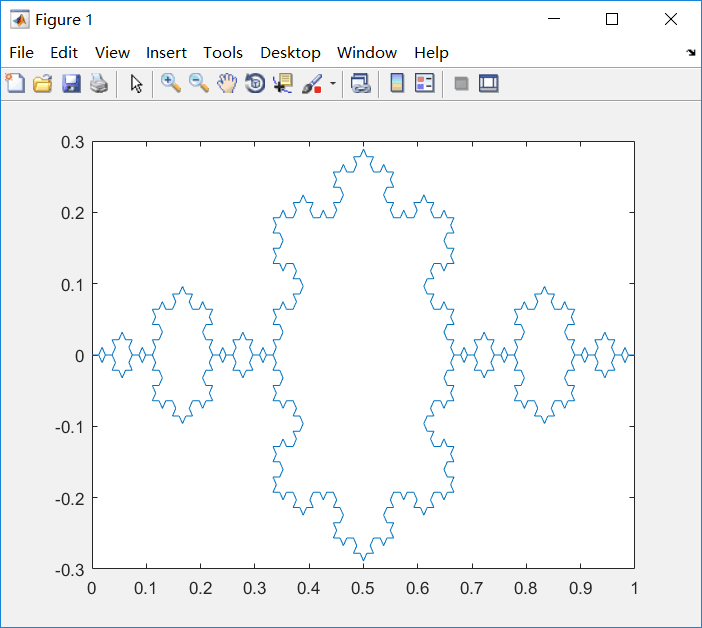
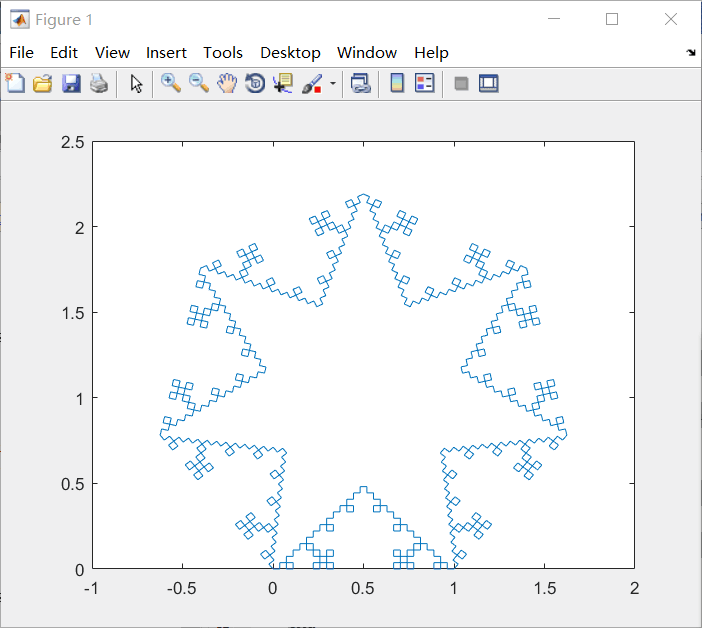
else

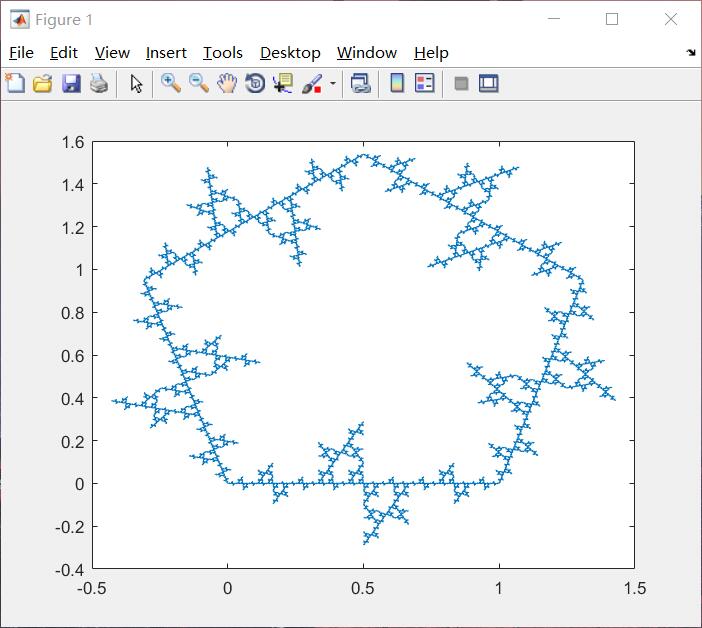
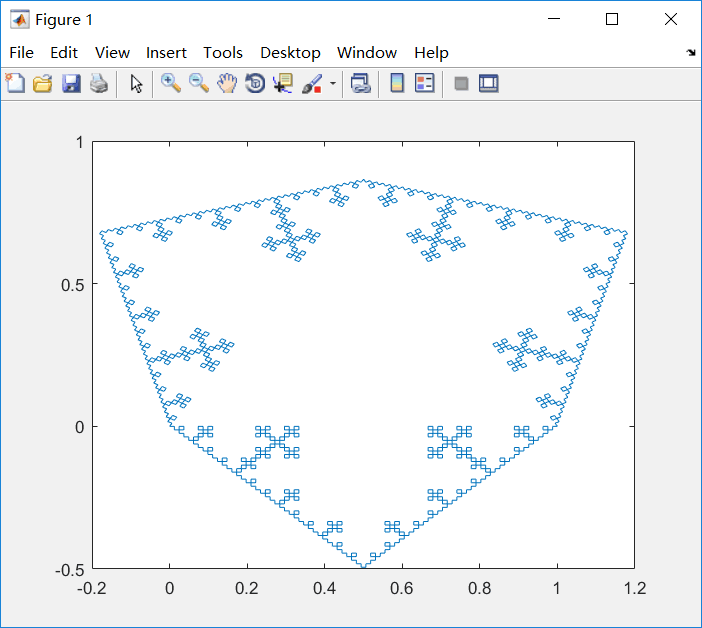
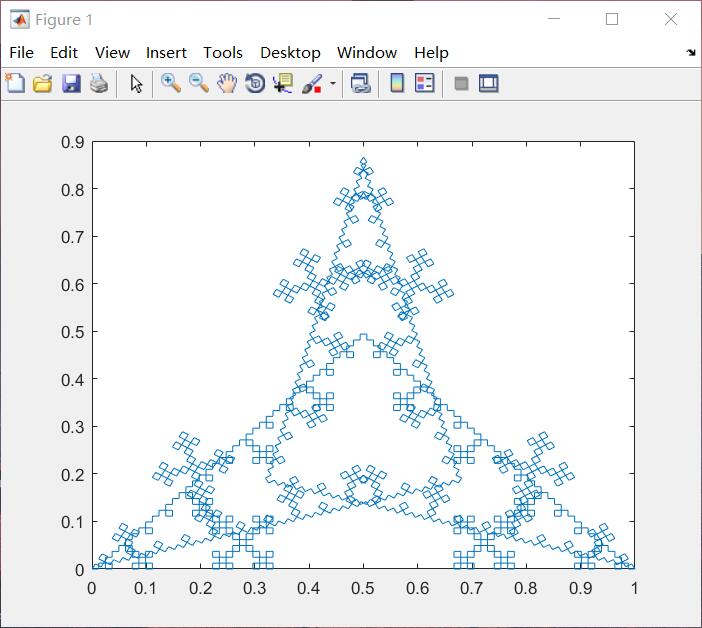
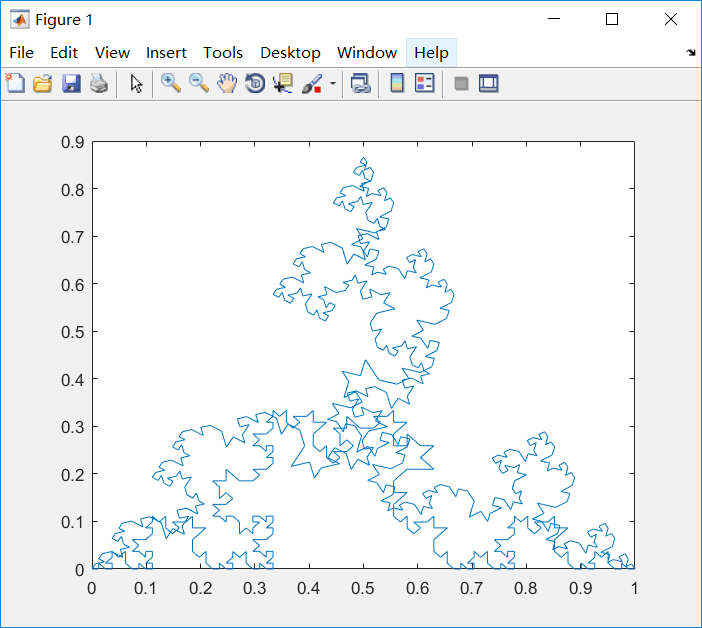
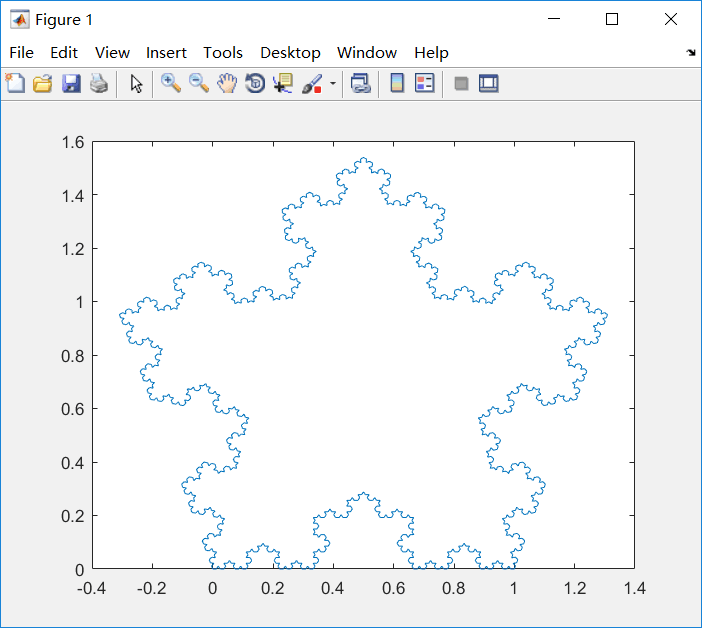
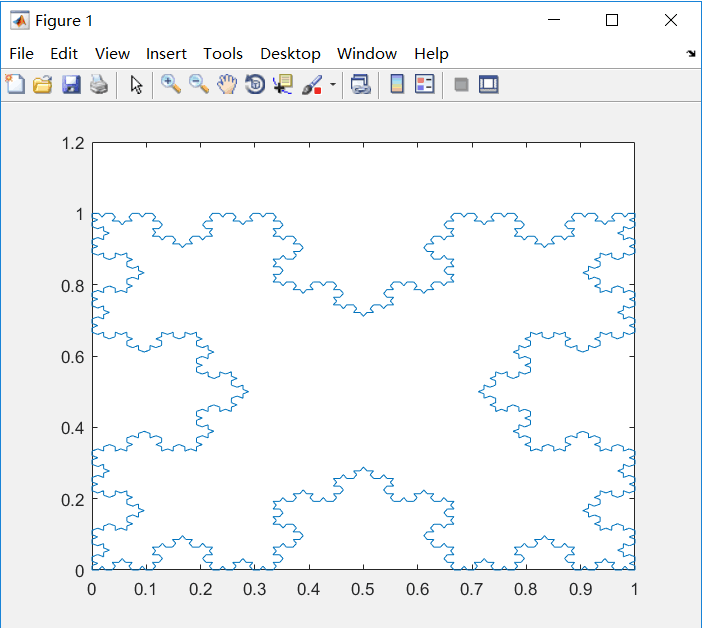
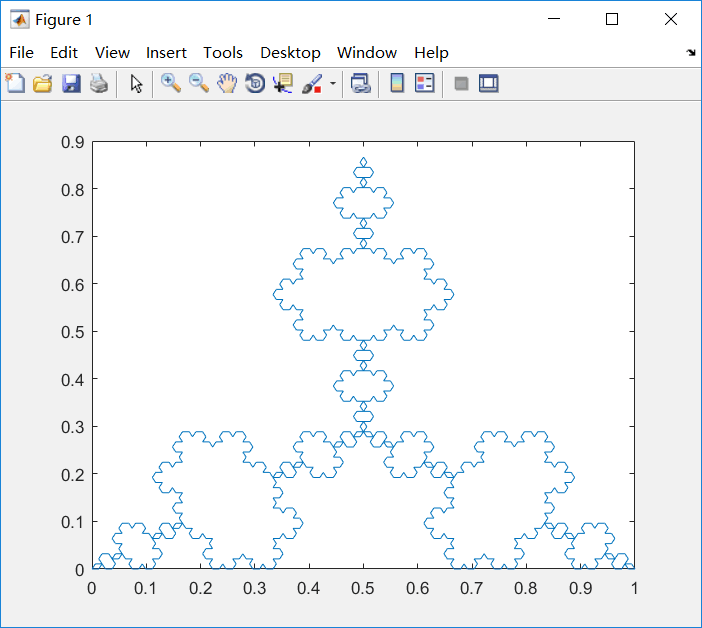
p=[p(:,1:i),p3,p5,p4,p(:,i+1:end)];

end

end

end

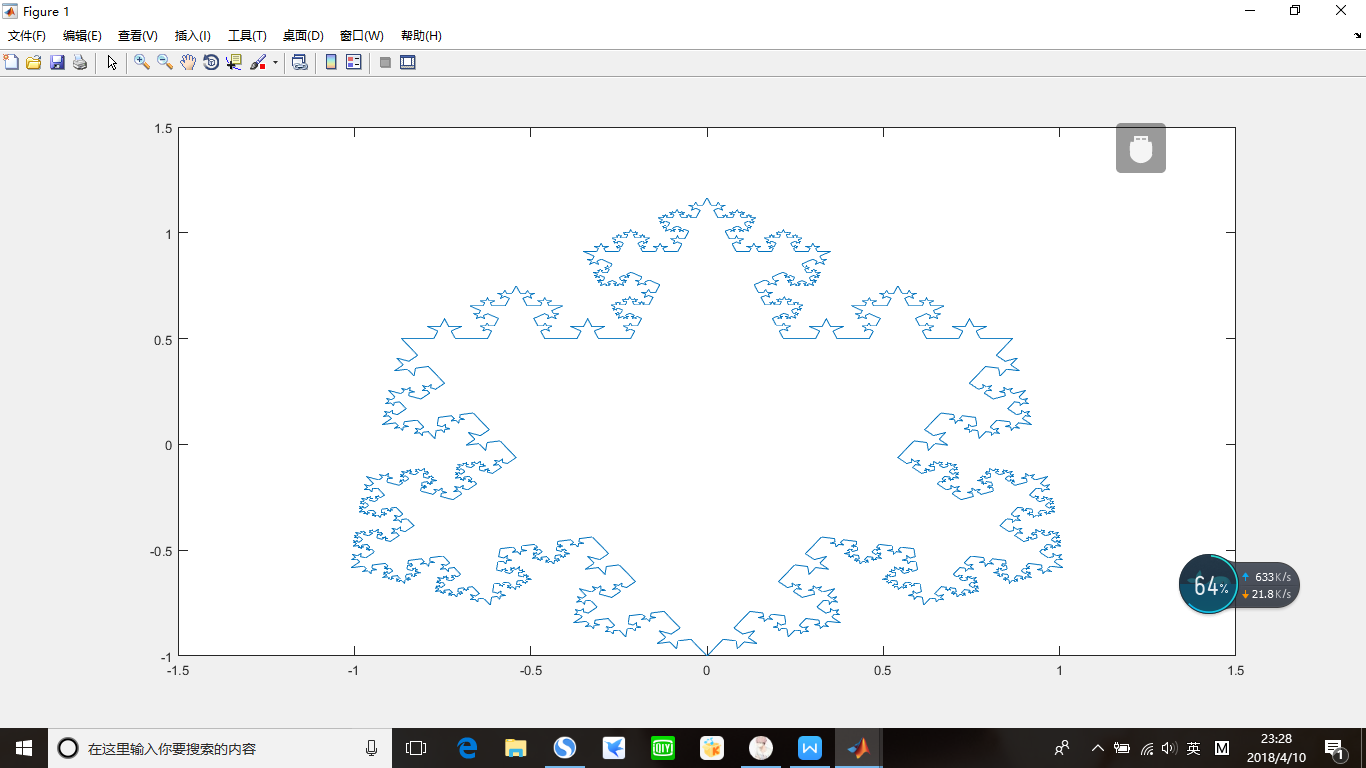
plot(p(1,:),p(2,:))



**以上为同一段代码部分功能的展示**

**4.三角形上生成“五角星”**

ratio1=2+0.5/cos(pi/5);  
ratio2=ratio1-1;  
r1=[cos(2.\*pi/5) -sin(2.\*pi/5);sin(2.\*pi/5) cos(2.\*pi/5)];  
r2=[cos(2.\*pi/5) sin(2.\*pi/5);-sin(2.\*pi/5) cos(2.\*pi/5)];  
r3=[cos(pi/5) -sin(pi/5);sin(pi/5) cos(pi/5)];  
p=[cos(pi/6) 0 -cos(pi/6) cos(pi/6);sin(pi/6) -1 sin(pi/6) sin(pi/6)];  
for n=1:3  
    q=p;  
    p=zeros(2,3.\*10^n+1);  
    for ii=1:3.\*10^(n-1)  
        x=q(:,ii+1)-q(:,ii);                         
        x1=3/8.\*x;  
x2=2/3.\*0.5/cos(pi/5).\*x1;  
p(:,10.\*ii-9)=q(:,ii);  
        p(:,10.\*ii-8)=q(:,ii)+x1;  
        p(:,10.\*ii-7)=q(:,ii)+x1+r1\*x2;  
        p(:,10.\*ii-6)=q(:,ii)+x1+r1\*x2+r1\*r1\*x2;  
        p(:,10.\*ii-5)=q(:,ii)+x1+ratio2.\*r1\*x2;  
        p(:,10.\*ii-4)=q(:,ii)+x1+ratio1.\*r1\*x2;  
        p(:,10.\*ii-3)=q(:,ii+1)-x1-ratio2.\*r2\*x2;  
        p(:,10.\*ii-2)=q(:,ii)+x1+ratio1.\*r3\*x2;  
        p(:,10.\*ii-1)=q(:,ii)+x1+ratio2.\*r3\*x2;  
        p(:,10.\*ii)=q(:,ii+1)-x1;  
        end  
p(:,3.\*10^n+1)=[cos(pi/6) sin(pi/6)]';  
end  
x1=p(1,:);  
y1=p(2,:);  
plot(x1,y1)



**5.GUI界面分形曲线**

function varargout = koch3(varargin)

gui\_Singleton = 1;

gui\_State = struct('gui\_Name', mfilename, ...

'gui\_Singleton', gui\_Singleton, ...

'gui\_OpeningFcn', @koch3\_OpeningFcn, ...

'gui\_OutputFcn', @koch3\_OutputFcn, ...

'gui\_LayoutFcn', [] , ...

'gui\_Callback', []);

if nargin && ischar(varargin{1})

gui\_State.gui\_Callback = str2func(varargin{1});

end

if nargout

[varargout{1:nargout}] = gui\_mainfcn(gui\_State, varargin{:});

else

gui\_mainfcn(gui\_State, varargin{:});

end

function koch3\_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;

guidata(hObject, handles);

function varargout = koch3\_OutputFcn(hObject, eventdata, handles)

varargout{1} = handles.output;

function edit1\_Callback(hObject, eventdata, handles)

cishu=get(hObject,'String');

guidata(hObject,handles);

function edit1\_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))

set(hObject,'BackgroundColor','white');

end

function pushbutton2\_Callback(hObject, eventdata, handles)

cishu=get(handles.edit1,'String');

ci=str2num(cishu);

pp=[cos(pi/2),-sin(pi/2);sin(pi/2) cos(pi/2)];

p=[0 1 1 0 0;0 0 1 1 0];

for j=1:ci

n=size(p,2);

for i=1:4:4\*(n-1)

p1=p(:,i);

p2=p(:,i+1);

p3=p1+(p2-p1)/3;

p4=2\*p3-p1;

p5=p3+pp\*(p4-p3);

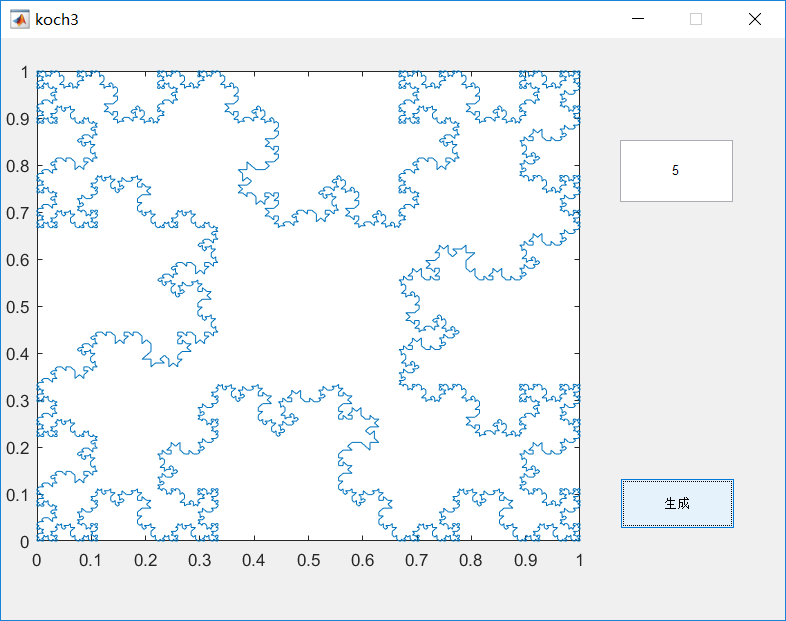
p=[p(:,1:i),p3,p5,p4,p(:,i+1:end)];

end

end

plot(p(1,:),p(2,:));

guidata(hObject,handles);



**6.人员分工：**

编程拓展：郭志猛25%

编程拓展：漆宇豪25%

编程及成果总结：陈天翔25%

编程及成果总结：邵戈文25%