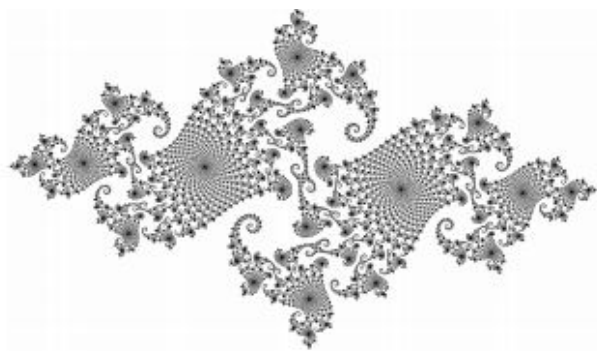
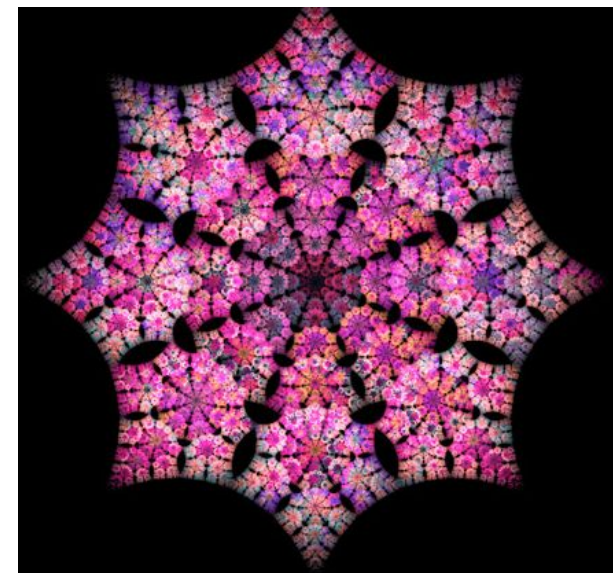
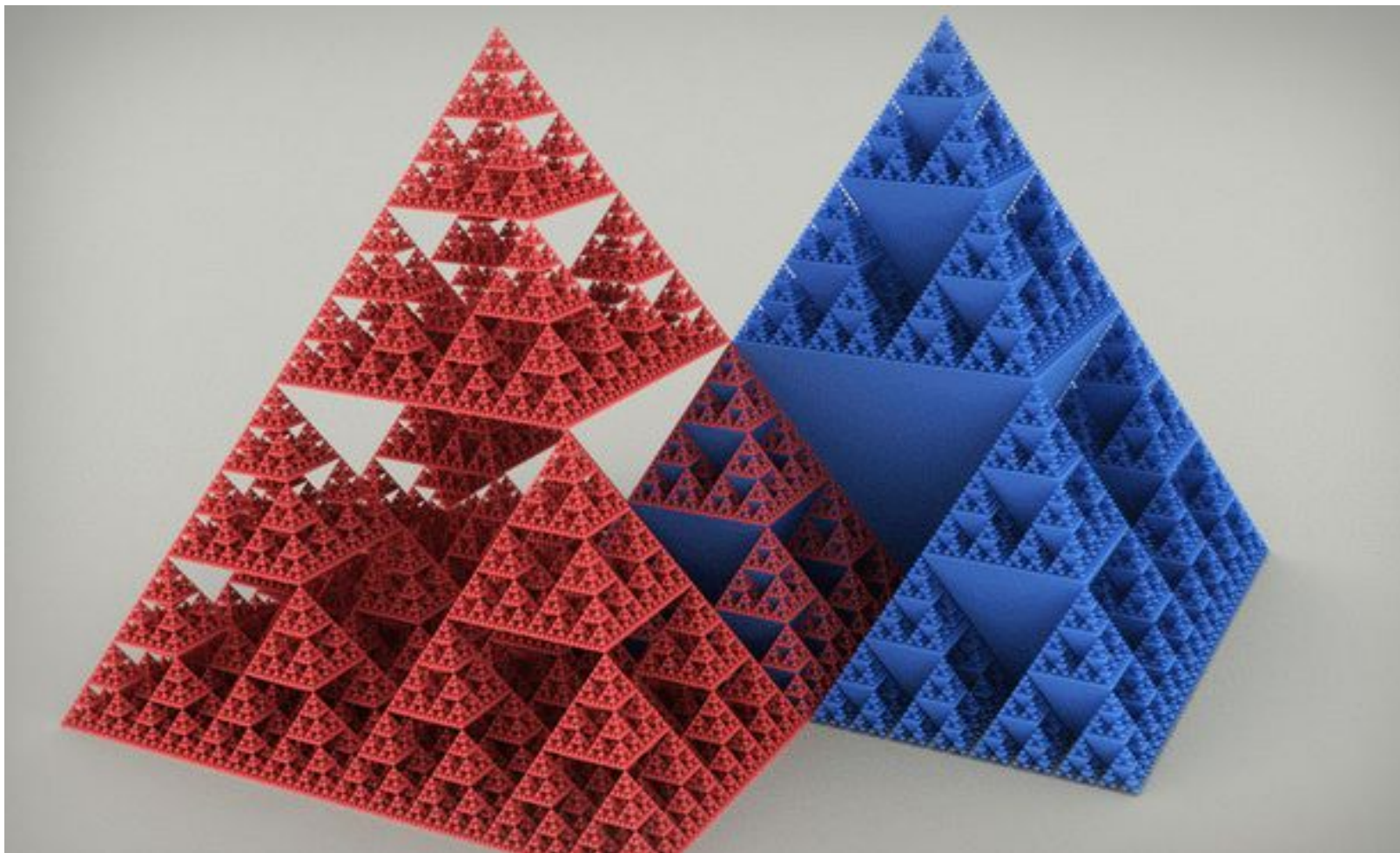



МОДЕЛИРОВАНИЕ  
ФРАКТАЛОВ В СИСТЕМЕ  
MAXIMA

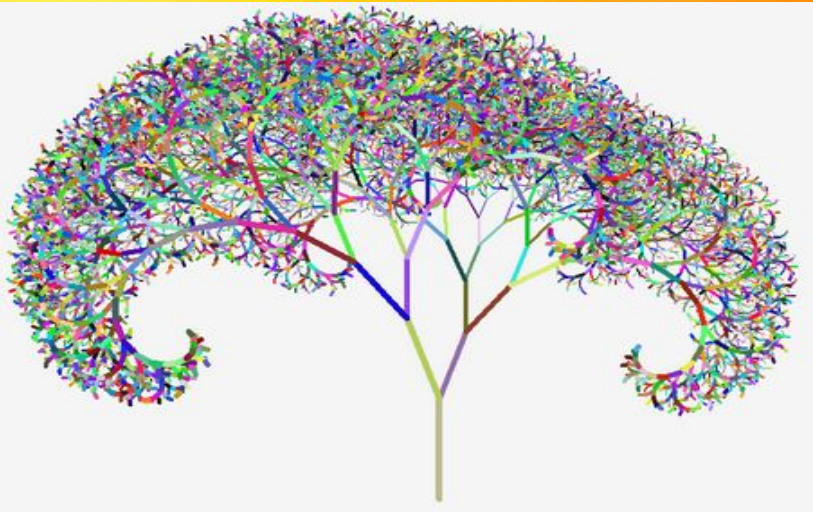


# ЧТО ТАКОЕ ФРАКТАЛЫ

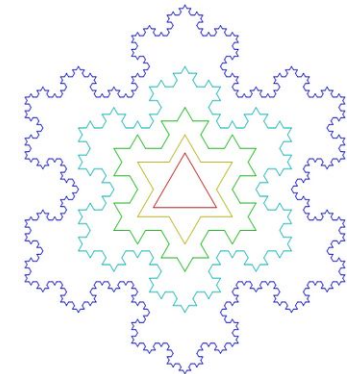
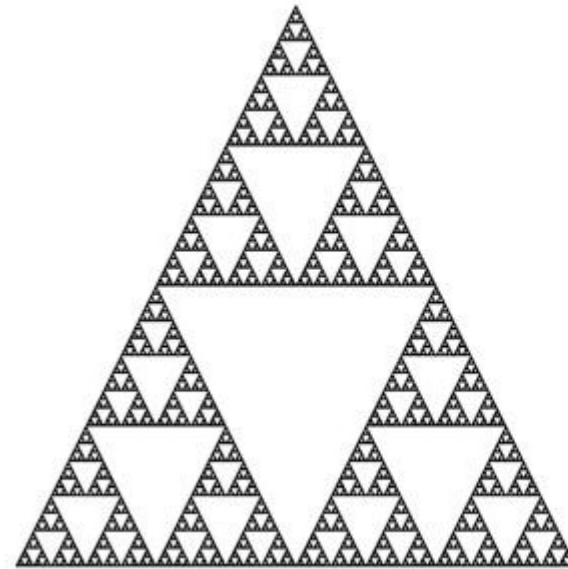


- 
- Обладает сложной структурой при любом увеличении;
  - Является (приближенно) самоподобной;
  - Обладает дробной хаусдорфовой (фрактальной) размерностью, которая больше топологической;
  - Может быть построена рекурсивными процедурами

# ОБЗОР ПАКЕТА FRACTALS



- треугольник Серпинского, фракталы «Дерево», «Папоротник»;
- множество Мандельброта и множества Жюлиа;
- снежинки Коха;
- отображения Пеано: кривые Серпинского и Гильберта.

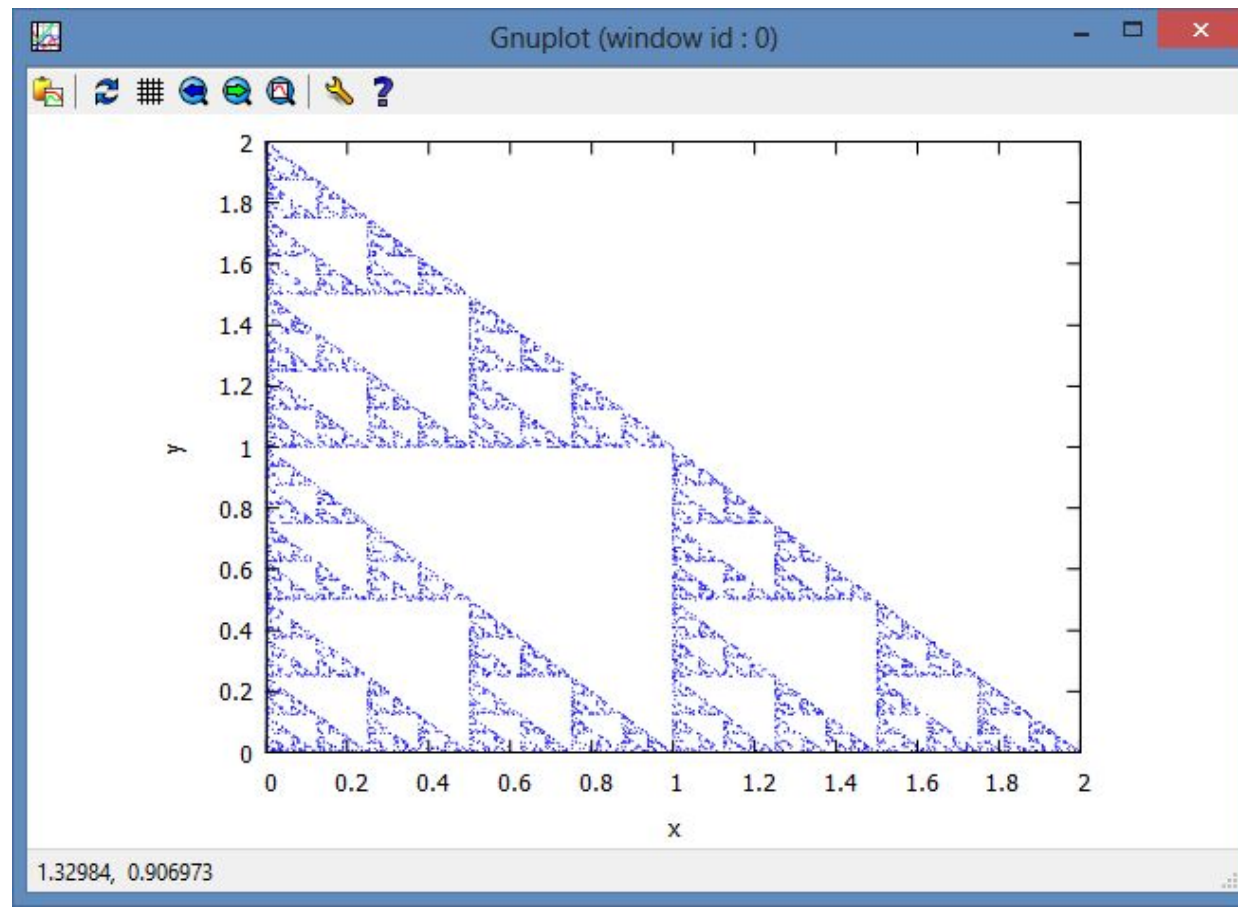


# ФУНКЦИИ ПАКЕТА FRACTALS

- ☆ `sierpinski(n)`
- ☆ `tree(n)`
- ☆ `fern(n)`
- ☆ `mandelbrot_set(x, y)`
- ☆ `julia_set(x, y)`
- ☆ `julia_sin(x, y)`
- ☆ `snowmap(vert, iter)`
- ☆ `hilbertmap(iter)`
- ☆ `sierpinski_map(iter)`

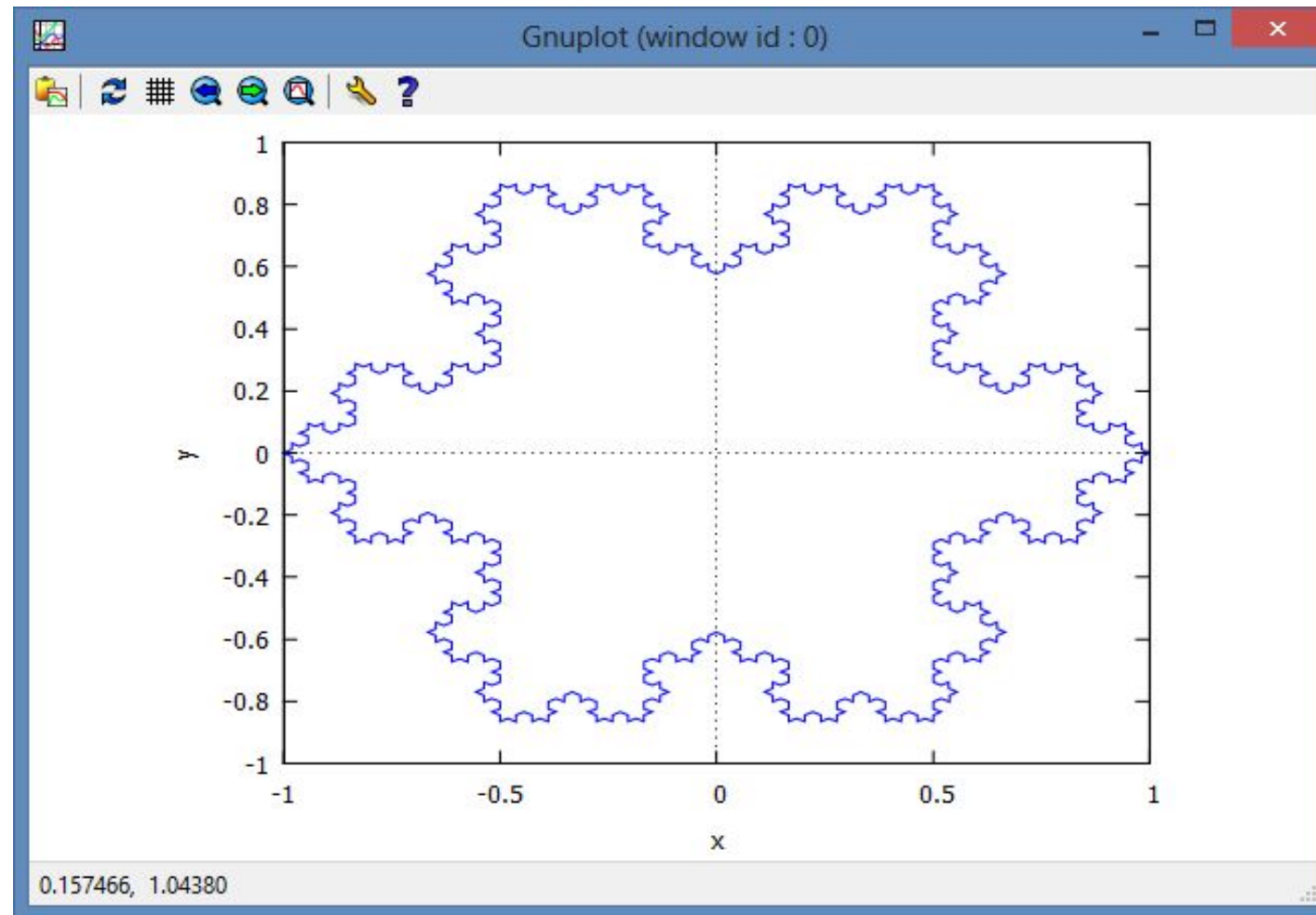
# ПРИМЕРЫ

```
(%i1) load(fractals)$  
plot2d([discrete,sierpinski(10000)], [style,dots])$
```



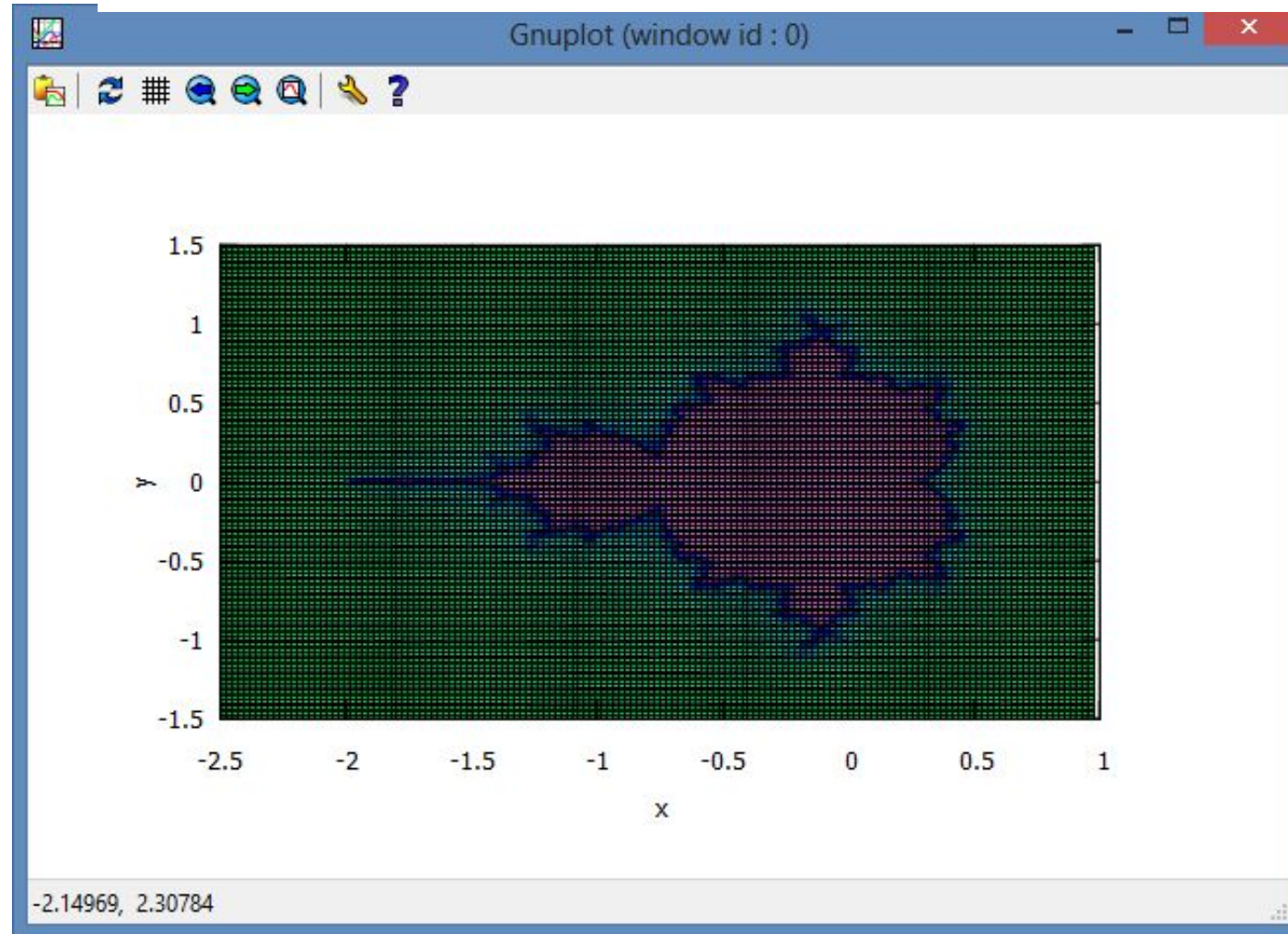
# ПРИМЕРЫ

```
(%i5) load(fractals)$  
plot2d([discrete,snowmap([1,exp(-%i*%pi*2/3),exp(%i*%pi*2/3),1],4)])$
```



# ПРИМЕРЫ

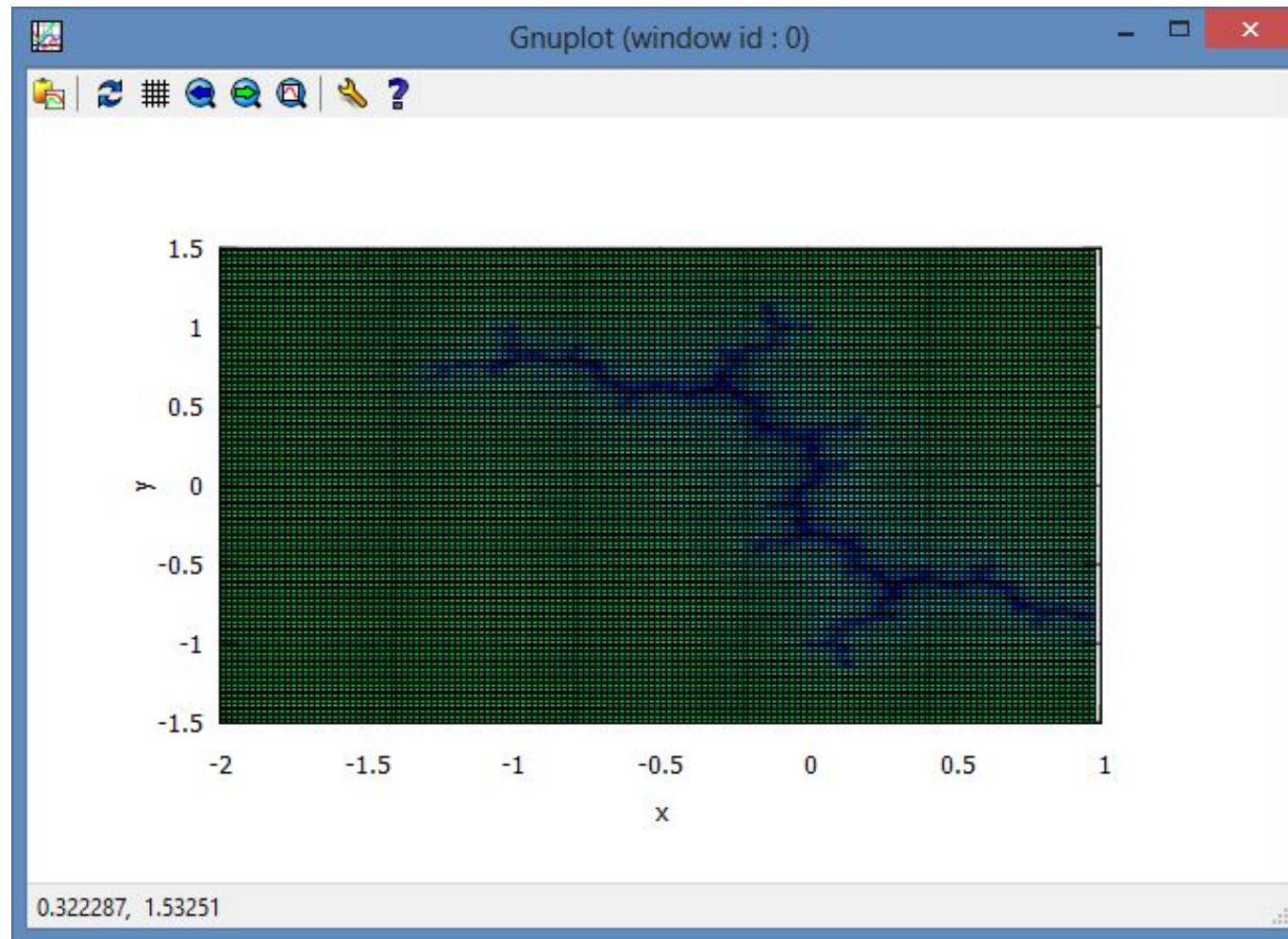
```
(%i3) load(fractals)$  
plot3d(mandelbrot_set, [x, -2.5, 1], [y, -1.5, 1.5],  
[gnuplot_preamble, "set view map"], [grid, 150, 150])$
```



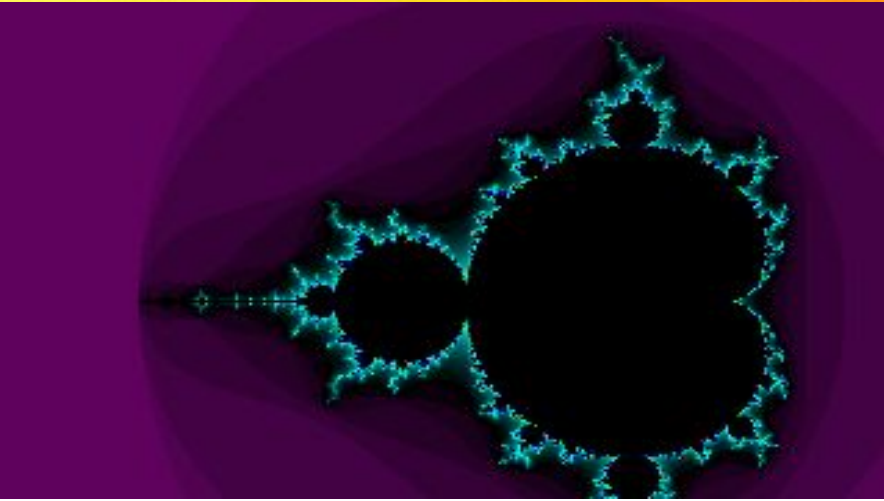


# ПРИМЕРЫ

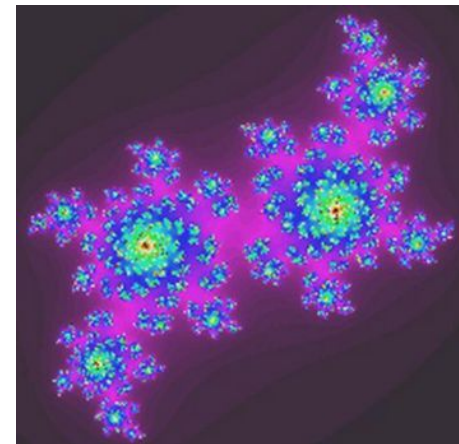
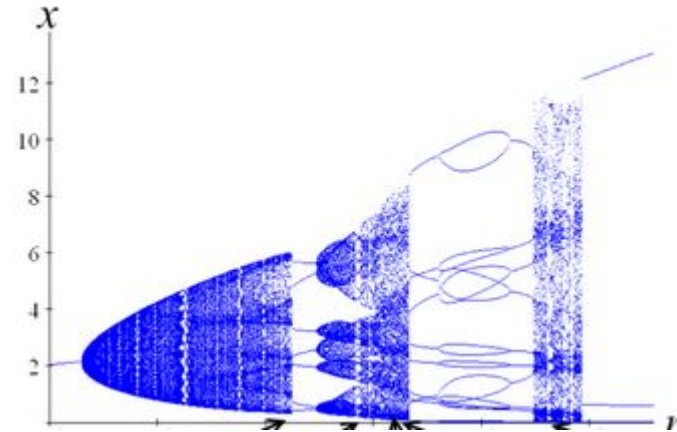
```
(%i1) load(fractals)$  
plot3d(julia_set, [x, -2, 1], [y, -1.5, 1.5],  
[gnuplot_preamble, "set view map"], [grid, 150, 150])$
```



# ОБЗОР ПАКЕТА DYNAMICS



- паутинная диаграмма;
- бифуркационная диаграмма;
- эволюция орбиты одно- и двумерного отображений;
- «игра в хаос»;
- система итерированных функций, заданная аффинными преобразованиями;
- множества Жюлиа, Мандельброта;

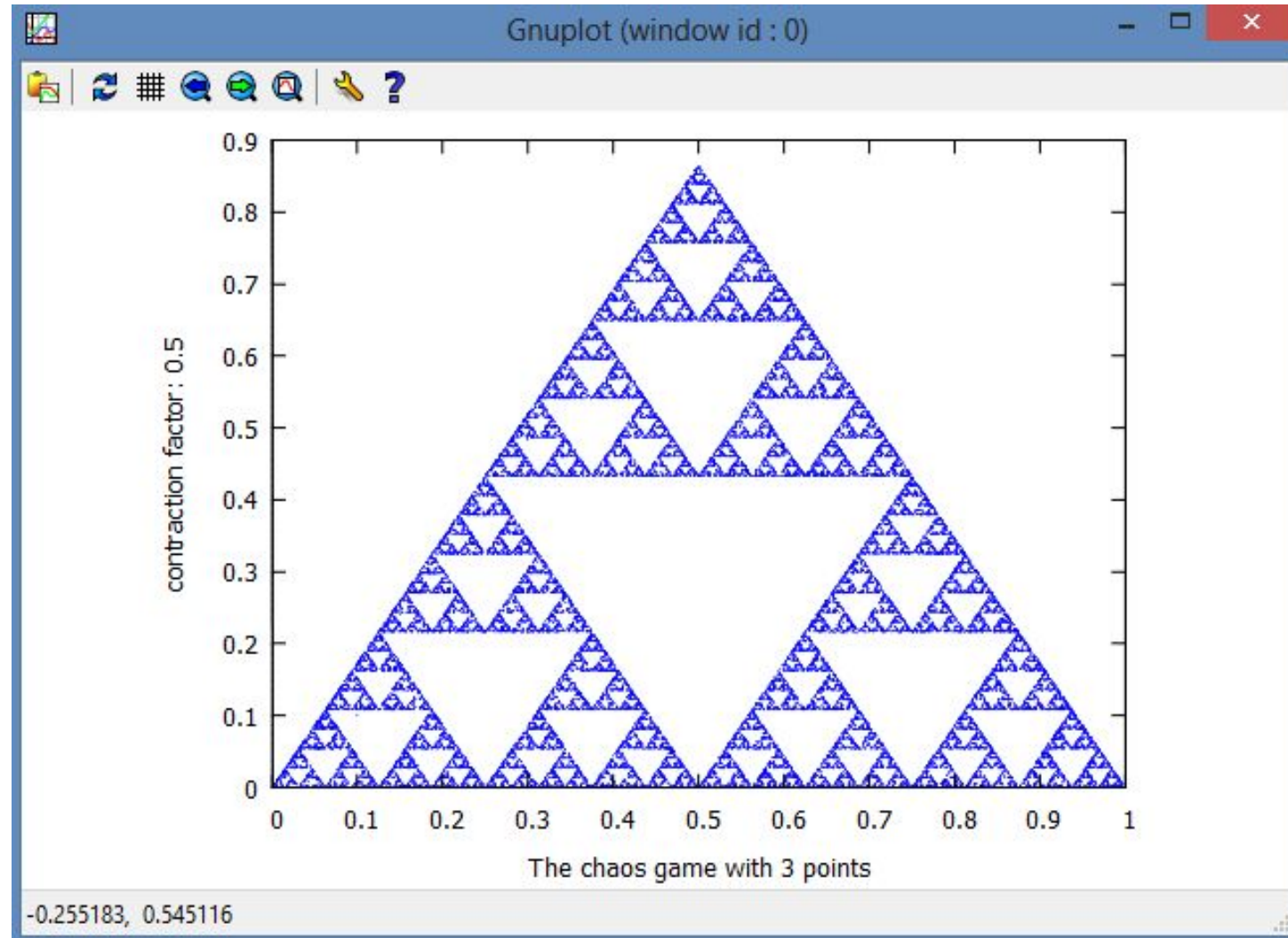


# ФУНКЦИИ ПАКЕТА DYNAMICS

- ★ `chaosgame([[x1, y1], ..., [xm, ym]], [x0, y0], b, n, options)`
- ★ `ifs([r1, ..., rm], [A1, ..., Am], [[x1, y1], ..., [xm, ym]], [x0, y0], n, options)`
- ★ `julia(x, y, options)`
- ★ `mandelbrot(options)`

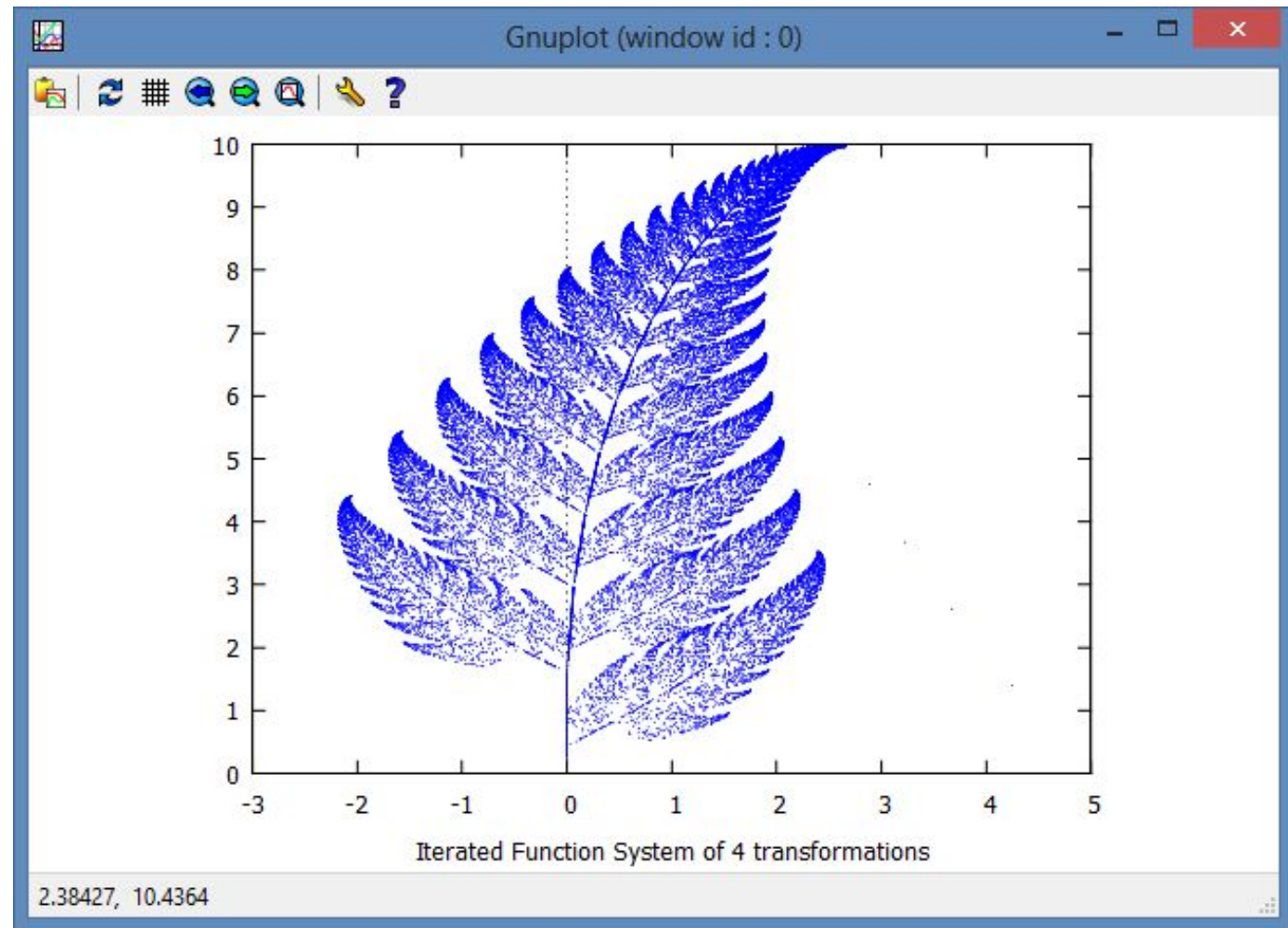
# «ИГРА В ХАОС»

```
(%i1) load(dynamics)$  
chaosgame([[0,0],[1,0],[0.5,sqrt(3)/2]],  
[0.1,0.1],1/2,30000,[style,dots])$
```



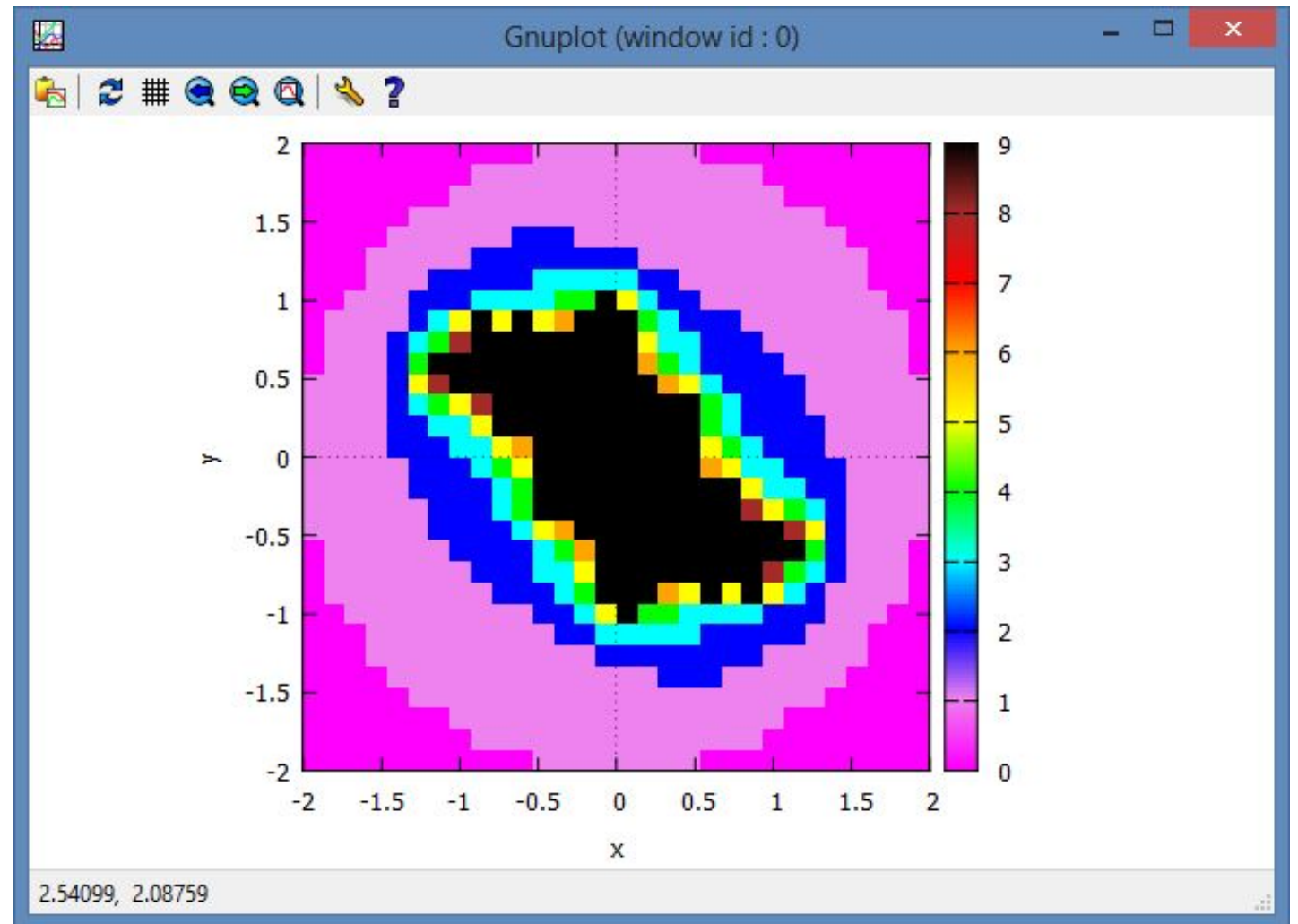
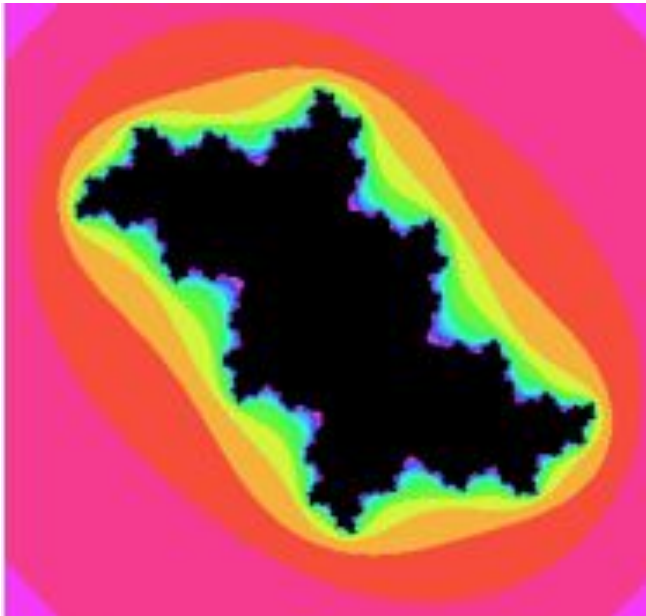
# ПОСТРОЕНИЕ АТТРАКТОРА СИСТЕМЫ ИТЕРИРОВАННЫХ ФУНКЦИЙ

```
(%i1) load(dynamics)$  
a1:matrix([0.85,0.04],[-0.04,0.85])$  
a2:matrix([0.2,-0.26],[0.23,0.22])$  
a3:matrix([-0.15,0.28],[0.26,0.24])$  
a4:matrix([0,0],[0,0.16])$  
p1:[0,1.6]$  
p2:[0,1.6]$  
p3:[0,0.44]$  
p4:[0,0]$  
r:[85,92,99,100]$  
ifs(r,[a1,a2,a3,a4],[p1,p2,p3,p4],  
[5,0],50000,[style,dots])$
```



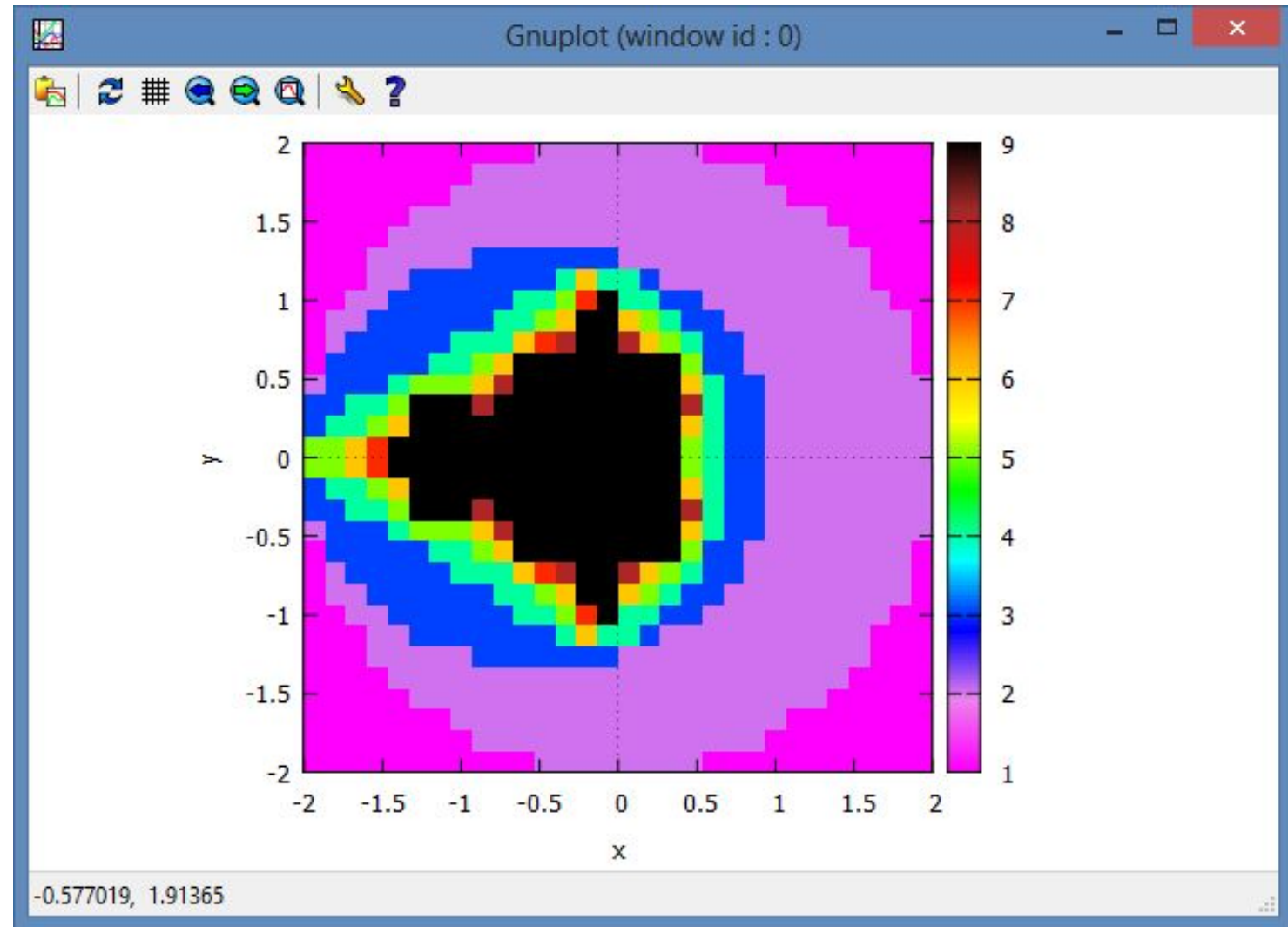
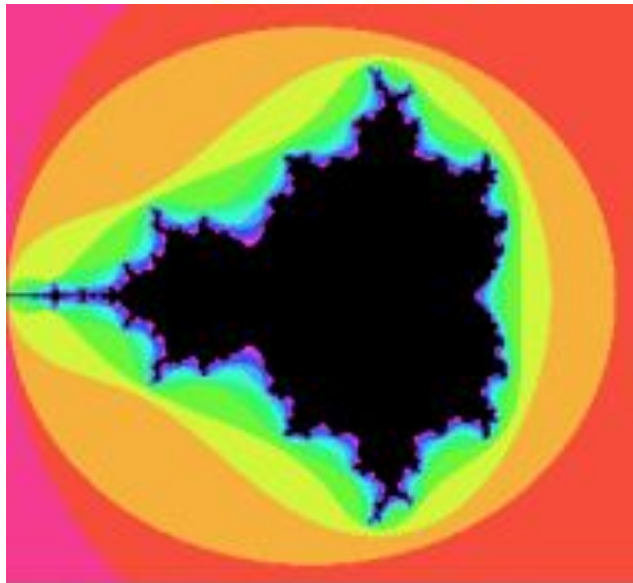
# МНОЖЕСТВО ЖЮЛИА

```
(%i14) load(dynamics)$  
julia(-0.11,0.65569999)$
```



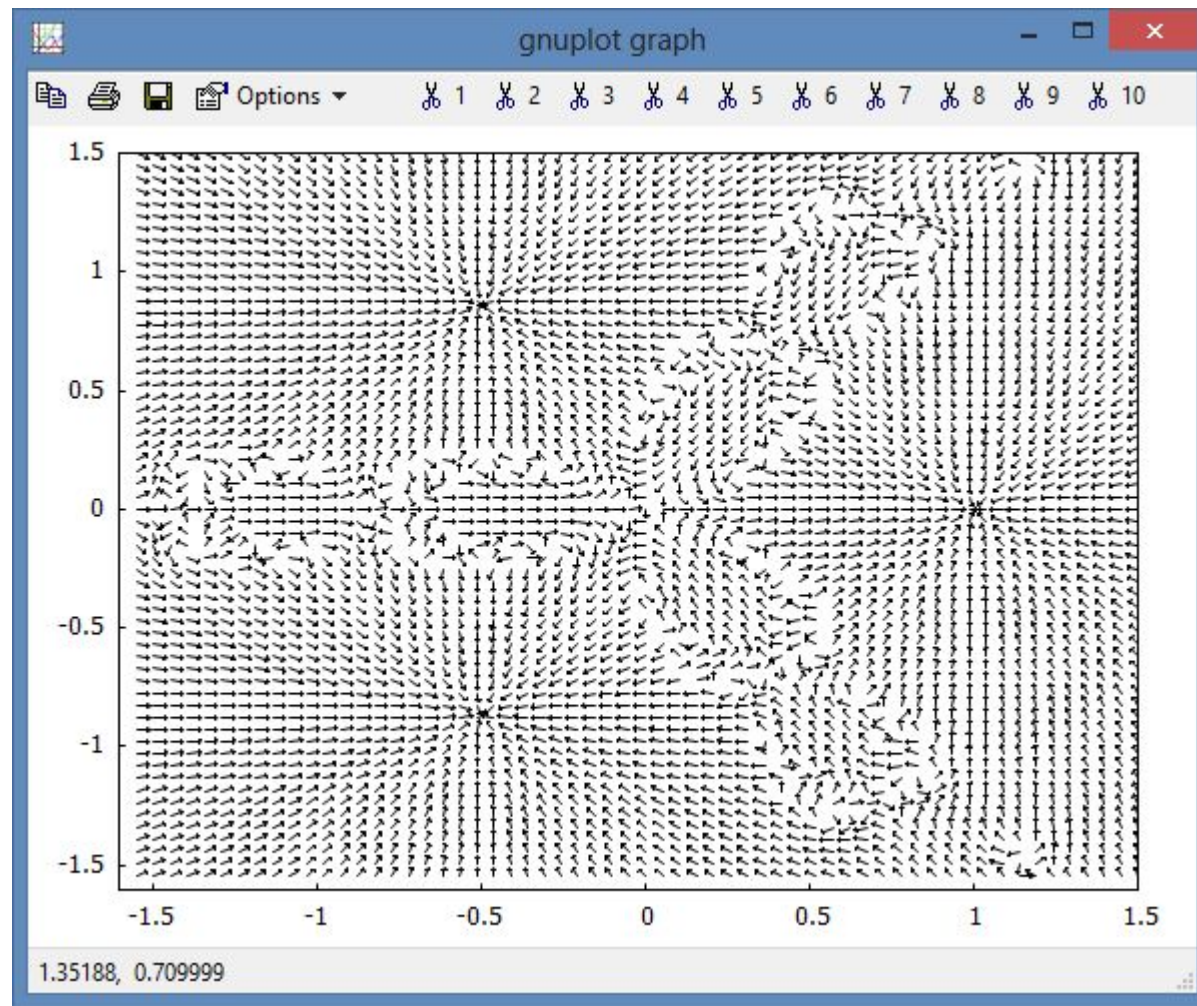
# МНОЖЕСТВО МАНДЕЛЬБРОТА

```
(%i16) load(dynamics) $  
mandelbrot() $
```



# ВЕКТОРНОЕ ПОЛЕ ДЛЯ ПРОБЛЕМЫ КЭЛИ (случай $f(z) = z^3 - 1$ )

```
(%i7) n:60$  
m:60$  
[a,b]:[-1.6,1.5]$  
[c,d]:[-1.6,1.5]$  
ITER:5$  
g(z):=z-(z^3-1)/(3*z^2),float$  
Vectors:[]$  
  
for i:1 thru n do  
for j:1 thru m do  
(z0:float(a+(b-a)*i/n+(c+(d-c)*j/m)*%i),  
z:z0,  
thru ITER do z:g(z),  
A:[realpart(z-z0),imagpart(z-z0)],  
len:sqrt(A[1]^2+A[2]^2)/0.04,  
Vectors:append(Vectors,[vector(float([a+(b-a)*i/n,c+(d-c)*j/m]),A/len))  
)$  
load(draw)$  
draw2d(color=black,xrange=[a,b],yrange=[c,d],head_length = 0.01,Vectors)$
```





- 
- [http://kpfu.ru/docs/F1416066913/main.\\_1\\_.pdf](http://kpfu.ru/docs/F1416066913/main._1_.pdf)