

ЛАБОРАТОРНА РОБОТА №3

РЕКУРСИВНИЙ АЛГОРИТМ ГЕНЕРАЦІЇ ФРАКТАЛІВ

Фрактал (лат. *fractus* — дроблений, сломанный, разбитый) — термин, означающий геометрическую фигуру, обладающую свойством самоподобия, то есть составленную из нескольких частей, каждая из которых подобна всей фигуре целиком.

Цель: изучение основ построения фрактальных множеств с использованием рекурсивных алгоритмов.

Изучение алгоритма, основанного на использовании рекурсивной функции, проведем на примере построения простого самоподобного фрактала — ковра Серпинского. В рассматриваемом алгоритме используется способ построения, основанный на последовательном удалении из начальной области внутренних подобластей в соответствии с заданными правилами. Выберем в качестве начального множества $0 S$ равносторонний треугольник вместе с областью, которую он замыкает. Удалим внутренность центральной треугольной области и назовем оставшееся множество $1 S$ (рис. 1). Затем повторим описанный процесс для каждого из трех оставшихся треугольников и получим приближение $2 S$. Продолжая таким образом, получим последовательность вложенных множеств $n S$, пересечение которых и образует ковер Серпинского S (рис. 1). Из построения видно, что ковер является объединением $N = 3$ существенно непересекающихся уменьшенных в два раза копий (коэффициенты подобия по горизонтали и вертикали в данном случае оказываются одинаковыми, $1/r = 2$). Фрактальная размерность ковра Серпинского d

$$d = \frac{\log(3)}{\log(2)} \approx 1.5850 \quad (1)$$

Для построения рассматриваемого фрактала, очевидно, можно использовать следующий алгоритм.

1. Задать порядок ковра N .
2. Задать координаты вершин исходного треугольника

$$ABC: (X_A, Y_A), (X_B, Y_B), (X_C, Y_C)$$

3. Построить равносторонний треугольник ABC и залить его синим цветом.
4. Вычислить координаты середин сторон треугольника ABC :

$$dx = \frac{X_B - X_A}{2}, \quad dy = \frac{Y_B - Y_A}{2},$$

$$X_{A'} = X_A + dx, Y_{A'} = Y_A + dy,$$

$$X_{B'} = X_B + dx + \frac{dx}{2}, Y_{B'} = Y_B + dy,$$

$$X_{C'} = X_C + \frac{dx}{2}, Y_{C'} = Y_C + dy$$

5. Построить треугольник $A'B'C'$ и залить его белым цветом.

6. Повторить N раз действия, описанные в пп. 4, 5, для треугольников $AA'C'$, $A'BB'$, $C'B'C'$, соответственно.

Наиболее просто описанный выше алгоритм можно реализовать при использовании рекурсивной процедуры, выполняющей последовательность действий, описанных в пп. 4, 5. Для реализации алгоритма в пакете MATLAB следует создать специальную функцию, возвращающую изображение ковра Серпинского, используя для этого встроенный текстовый редактор MATLAB или любой другой текстовый редактор (например, «Блокнот»), и сохранить текст в файле *Serpinsky.m*.

Для вывода изображения ковра Серпинского, например, пятого порядка, следует ввести в командной строке пакета MATLAB имя функции с соответствующим значением:

>> *Serpinsky*(5);

Результат, возвращаемый функцией *Serpinsky()*, показан на рис. 1 (множество S_5).

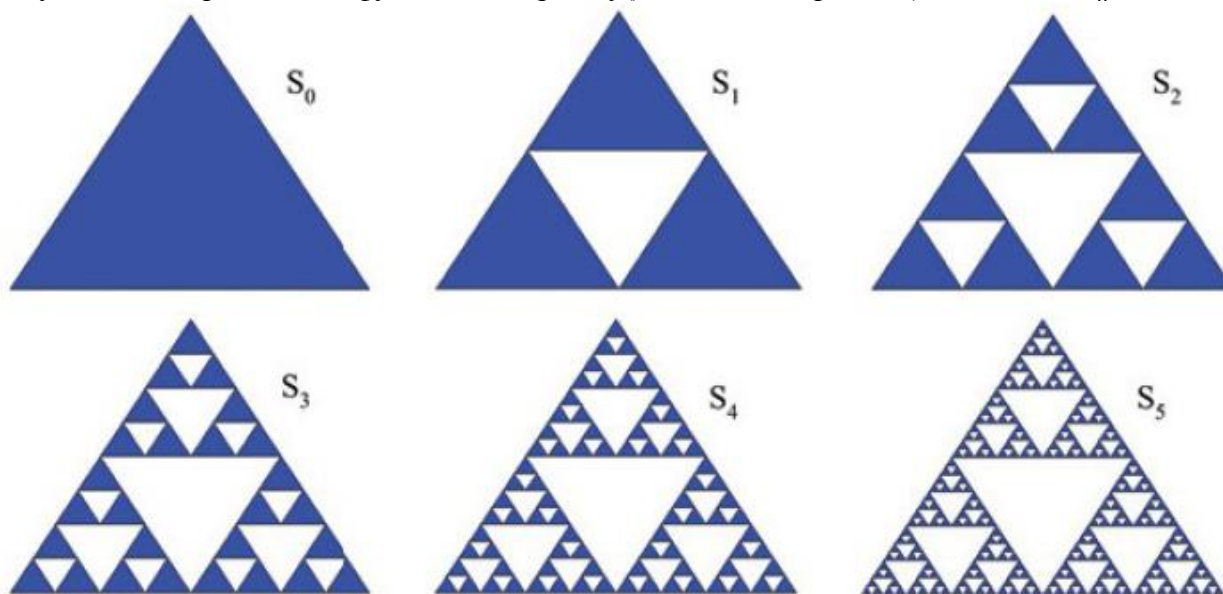


Рис. 1 – Построение ковра Серпинского

Описанный выше алгоритм легко обобщается для фрактальных объектов, правила построения которых аналогичны правилам построения треугольного ковра Серпинского. Например, алгоритм визуализации фрактального объекта, основные этапы построения которого представлены на рис. 2, реализуется следующей последовательностью действий.

1. Задать порядок ковра N .
2. Задать координаты вершин исходного квадрата

$$ABCD: (X_A, Y_A), (X_B, Y_B), (X_C, Y_C), (X_D, Y_D)$$

3. Построить квадрат $ABCD$ и залить его синим цветом.
4. Вычислить координаты точек, делящих стороны квадрата $ABCD$ на три равные части:

$$dx = \frac{X_B - X_A}{3}, \quad dy = \frac{Y_B - Y_A}{3}$$

$$X_{A'} = X_A + dx, \quad Y_{A'} = Y_A + dy,$$

$$X_{B'} = X_A + dx + dx, \quad Y_{B'} = Y_A + dy + dy,$$

$$X_{C'} = X_A + dx + dx + dx, \quad Y_{C'} = Y_A + dy + dy + dy,$$

$$X_{D'} = X_A + dx, \quad Y_{D'} = Y_A + dy + dy + dy$$

5. Построить квадрат $A'B'C'D'$ и залить его белым цветом.

6. Повторить N раз действия, описанные в пп. 4, 5, для квадратов с вершинами, имеющими следующие координаты:

$$\begin{aligned} &(X_A, Y_A), (X_{A'}, Y_{A'}), (X_{A'}, Y_{A'}), (X_A, Y_{A'}), \\ &(X_{A'}, Y_{A'}), (X_{B'}, Y_{B'}), (X_{B'}, Y_{B'}), (X_{A'}, Y_{B'}); \\ &(X_{B'}, Y_{A'}), (X_B, Y_B), (X_B, Y_{B'}), (X_{B'}, Y_{B'}), \\ &(X_{B'}, Y_{B'}), (X_B, Y_{B'}), (X_B, Y_{C'}), (X_{C'}, Y_{C'}); \\ &(X_{C'}, Y_{C'}), (X_B, Y_{C'}), (X_C, Y_C), (X_{C'}, Y_{C'}), \\ &(X_{D'}, Y_{D'}), (X_{C'}, Y_{C'}), (X_{C'}, Y_{C'}), (X_{D'}, Y_{C'}); \\ &(X_A, Y_{D'}), (X_{D'}, Y_{D'}), (X_{D'}, Y_C), (X_D, Y_D), \\ &(X_A, Y_{A'}), (X_{A'}, Y_{D'}), (X_{D'}, Y_{D'}), (X_A, Y_{D'}) \end{aligned}$$

соответственно.

Ниже приведен листинг файла Serpinsky2.m, который содержит описание функции, возвращающей изображение квадратного ковра Серпинского, представленного на рис. 2.

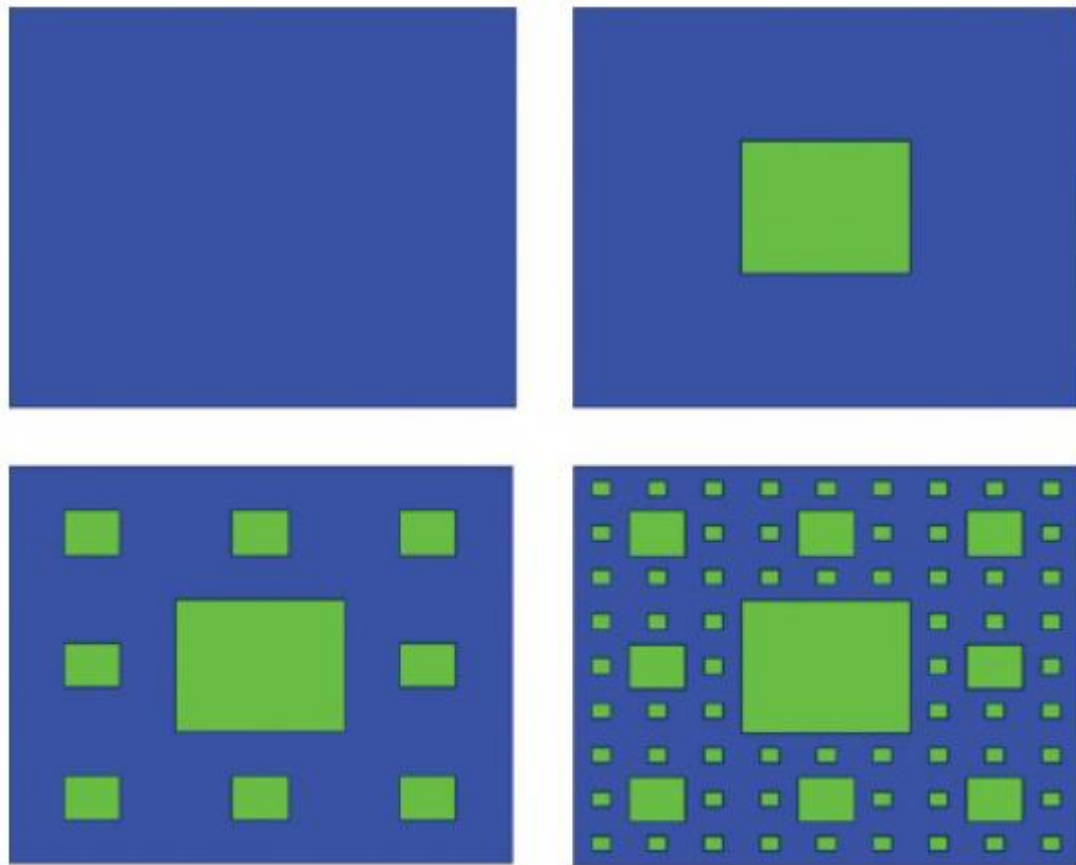


Рис. 2 – Этапы построения квадратного ковра Серпинского

Листинг файла **Serpinsky.m**

```
function z = Serpinsky(Lmax)
% функция, возвращающая изображение ковра Серпинского
% Lmax – порядок ковра
% задание координат вершин равнобедренного треугольника
x1=0; y1=0; x2=1; y2=0; x3=0.5; y3=sin(pi/3);
h=figure(1); % инициализация графического окна
hold on; % включение режима рисования фигур в одном
графическом окне
fill([x1 x2 x3],[y1 y2 y3],'k');
% прорисовка равностороннего треугольника
set(gca,'xtick',[],'ytick',[]); % отключение режима
оцифровки осей
set(gca,'XColor','w','YColor','w'); % установка цвета
рисования осей
Simplex(x1,y1,x2,y2,x3,y3,0,Lmax);
% обращение к функции, прорисовывающей равносторонние
треугольники
% белого цвета
hold off % отключение режима рисования фигур в одном
графическом окне
function z=Simplex(x1,y1,x2,y2,x3,y3,n,Lmax)
% рекурсивная функция, прорисовывающая равносторонние
треугольники
% белого цвета
if n<Lmax
% задание координат вершин текущего равностороннего
треугольника
```

```

dx=(x2-x1)/2; dy=(y3-y1)/2; x1n=x1+dx; y1n=y1;
x2n=x1+dx+dx/2;
y2n=y1+dy;x3n=x1+dx/2; y3n=y1+dy;
fill([x1n x2n x3n],[y1n y2n y3n],'w');
% прорисовка текущего равностороннего треугольника
n=n+1;
% рекурсия
Simplex(x1,y1,x1n,y1n,x3n,y3n,n,Lmax);
Simplex(x1n,y1n,x2,y2,x2n,y2n,n,Lmax);
Simplex(x3n,y3n,x2n,y2n,x3,y3,n,Lmax);
end

```

Листинг файла Serpinsky2.m

```

function z=Serpinsky2(Lmax)
% функция, возвращающая изображение квадратного ковра
Серпинского
% Lmax — порядок ковра
% Задание координат вершин исходного квадрата
x1=0; y1=0; x2=1; y2=0; x3=1; y3=1; x4=0; y4=1;
figure(1); hold on; fill([x1 x2 x3 x4],[y1 y2 y3 y4],'b');
set(gca,'xtick',[0 1],'ytick',[0 1]);
set(gca,'XColor','w','YColor','w');
Quadrante(x1,y1,x2,y2,x3,y3,x4,y4,0,Lmax);
hold off
function z=Quadrante(x1,y1,x2,y2,x3,y3,x4,y4,n,Lmax)
% рекурсивная функция, прорисовывающая квадраты белого
цвета
if n<Lmax
dx=(x2-x1)/3; dy=(y3-y1)/3;
x1n=x1+dx; y1n=y1+dy; x2n=x1+dx+dx; y2n=y1+dy;
x3n=x1+dx+dx; y3n=y1+dy+dy; x4n=x1+dx; y4n=y1+dy+dy;
fill([x1n x2n x3n x4n],[y1n y2n y3n y4n],'g');
n=n+1;
Quadrante(x1,y1,x1+dx,y1,x1+dx,y1+dy,x1,y1+dy,n,Lmax);
Quadrante(x1+dx,y1,x1+2*dx,y1,x1+2*dx,y1+dy,x1+dx,y1+dy,
,n,Lmax);
Quadrante(x1+2*dx,y1,x2,y1,x2,y1+dy,x1+2*dx,y1+dy,n,Lma
x);
Quadrante(x1+2*dx,y1+dy,x2,y1+dy,x2,y1+2*dy,x1+2*dx,y1+
2*dy,n,Lmax);
Quadrante(x1+2*dx,y1+2*dy,x2,y1+2*dy,x2,y3,x1+2*dx,y3,n
,Lmax);
Quadrante(x1+dx,y1+2*dy,x1+2*dx,y1+2*dy,x1+2*dx,y4,x1+d
x,y4,n,Lmax);
Quadrante(x1,y1+2*dy,x1+dx,y1+2*dy,x1+dx,y4,x1,y4,n,Lma
x);
Quadrante(x1,y1+dy,x1+dx,y1+dy,x1+dx,y1+2*dy,x1,y1+2*dy
,n,Lmax);
end

```

Еще одним примером фрактального объекта, для построения которого оказывается удобным использовать рекурсивный алгоритм, является кривая Коха. Построение данной кривой начинается с отрезка $0K$ единичной длины.

Удалим из отрезка K_0 отрезок длины $1/3$ и добавим два новых отрезка такой же длины, как показано на рис. 3. Назовем полученное множество K_1 . На следующем шаге разделим каждый отрезок длины $1/3$ на три части длины $1/9$ и повторим описанную процедуру, заменяя на каждом шаге среднюю ветвь двумя новыми отрезками. Обозначим через K_n фигуру, получившуюся после n -го шага. Можно строго доказать [6], что последовательность кривых

$$\{K_n\}_1^\infty$$

сходится к предельной кривой K бесконечной длины, фрактальная размерность которой равна

$$d = \frac{\log(4)}{\log(3)} \approx 1,2618$$

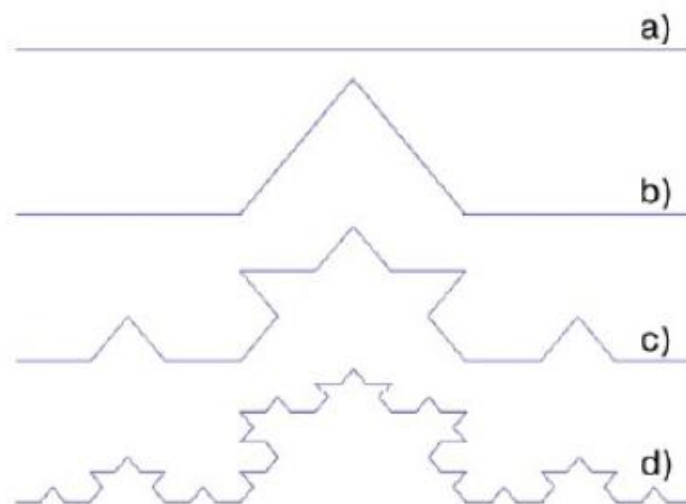


Рис. 3 – Построение кривой Коха: а) K_0 , б) K_1 , в) K_2 , д) K_3

Ниже приводится листинг рекурсивной функции, возвращающей изображение кривой Коха.

Листинг рекурсивной функции, возвращающей изображение кривой Коха

```
function z=Koch(N)
% функция, возвращающая изображение кривой Коха
x1=0; y1=0; % левая точка начального отрезка
x2=1; y2=0; % правая точка начального отрезка
figure(1); axis([0 1 0 1]);
set(gca,'XColor','w','YColor','w');
hold on;
Coord(x1,y1,x2,y2,N);
% вызов рекурсивной функции, прорисовывающей кривую Коха
function z=Coord(x1,y1,x2,y2,n)
% рекурсивная функция, прорисовывающая кривую Коха
if n>0
% вычисление координат концов отрезков на очередном
шаге рекурсии
dx=(x2-x1)/3; dy=(y2-y1)/3;
x1n=x1+dx; y1n=y1+dy;
x2n=x1+2*dx; y2n=y1+2*dy;
xmid=dx/2-dy*sin(pi/3)+x1n;
```

```

ymid=dy/2+dx*sin(pi/3)+y1n;
% рекурсия
Coord(x1, y1, x1n, y1n, n-1);
Coord(x1n, y1n, xmid, ymid, n-1);
Coord(xmid, ymid, x2n, y2n, n-1);
Coord(x2n, y2n, x2, y2, n-1);
else
r1=[x1 y1]; r2=[x2 y2]; R=cat(1, r1, r2);
plot(R(:,1),R(:,2),'Color','k'); % построение кривой
Коха
end;

```

Изображение кривой Коха пятого порядка K_5 , возвращенное описанной выше функцией, представлено на рис. 4.

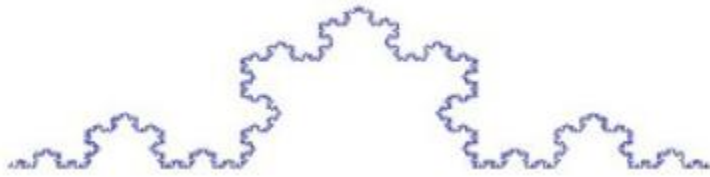


Рис. 4 – Кривая Коха пятого порядка

ЗАДАНИЕ

1. Изучить основы построения фрактальных объектов с помощью рекурсивных алгоритмов.
2. Построить треугольный ковёр Серпинского, базовым элементом для которого является не равносторонний, а прямоугольный треугольник.
3. Модифицировать кривую Коха, чтобы угол между центральными отрезками составлял не 60, а 90 градусов.

ЛИТЕРАТУРА

1. Данилов Ю. А., Кадомцев Б. Б. Что такое синергетика?//Нелинейные волны. Самоорганизация — М., Наука, 1983.
2. Каток А. Б., Хасселблат Б. Введение в современную теорию динамических систем.— М.: Факториал, 1999.
3. Лихтенберг А., Либерман М. Регулярная и стохастическая динамика.— М.: Меркурий_ПРЕСС, 2000.
4. Шустер Г. Детерминированный хаос.— М.: Мир, 1988.
5. Кренкель Э. Т. Сжатие сигналов с применением теории фракталов.— М.: МТУСИ, 1996.
6. Кроновер Р. М. Фракталы и хаос в динамических системах. Основы теории.— М.: Постмаркер, 2000.