МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ

ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

**институт информационных технологий и технологического образования**

**кафедра информационных технологий и электронного обучения**

Основная профессиональная образовательная программа

Направление подготовки 09.03.01 Информатика и вычислительная техника

Направленность (профиль) «Технологии разработки программного обеспечения»

форма обучения – очная

**Курсовая работа**

по дисциплине «Технологии компьютерного моделирования»

Фракталы и их моделирование

Обучающегося 2 курса

Нюхалова Дениса Глебовича

Руководитель:

к.п.н, доцент

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Гончарова С. В.

«\_\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

Санкт-Петербург

2020

СОДЕРЖАНИЕ

Оглавление

|  |  |
| --- | --- |
| Введение | 3 |
| Глава 1 Теоретическая часть |  |
| 1.1 Понятие фрактал | 4 |
| 1.2 Фрактальная размерность | 6 |
| 1.3 Применение фрактальной размерности | 12 |
| 1.4 Иные применения фракталов | 15 |
| Глава 2 Практическая часть |  |
| 2.1 Моделирование фракталов в растровых графических редакторах | 17 |
| 2.2 Моделирование фракталов при помощи языков программирования | 19 |
| Заключение | 25 |
| Источники | 26 |

ВВЕДЕНИЕ

Понятие фрактал в науке появилось относительно недавно. Фракталом можно назвать структуру, состоящую из частей, которые в каком-то смысле подобны целому. На самом деле это понятие намного шире, чем кажется. Если углубиться в понятие фрактала, то можно понять, что многие объекты в природе обладают его свойствами, например: побережья, облака, молнии, кроны деревьев, снежинки, ракушки, система кровообращения, альвеолы. Соответственно, для изучения и моделирования таких объектов можно воспользоваться свойствами и правилами фрактальной геометрии.

Фрактальная геометрия появилась благодаря Бенуа Мандельброту, выпустившему в 70 – 80 годах прошлого века ряд фундаментальных трудов, посвященных фракталам. Однако, как это бывает со многими математическими объектами, изучение фракталов невозможно без их моделирования. На самом деле, математическая база теории фракталов была заложена еще за много лет до Бенуа Мандельброта, однако сама теория смогла развиться лишь с появлением возможности представления фракталов при помощи вычислительных устройств.

Автор ставит своей целью рассмотреть азы теории фракталов, а также показать некоторые возможные способы моделирования фракталов при помощи современной вычислительной техники и программного обеспечения.

Теоретическая часть  
Параграф 1.1

Понятие фрактал

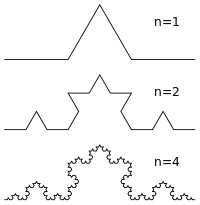
Для начала стоит определить, что можно называть фракталом. Это геометрическая фигура, которая:

* обладает сложной структурой при любом увеличении
* является самоподобной
* обладает дробной фрактальной размерностью, которая больше топологической
* построена рекурсивно

Фракталы можно условно подразделить на несколько типов:

Первый тип – геометрические фракталы. Для построения таковых нужно определить исходный паттерн, по которому будет строиться фрактал, зачастую это обычная ломаная линия, над которой будут совершаться какие либо действия. За один шаг алгоритма каждый из отрезков, составляющих ломаную, заменяется на ломаную-паттерн, в соответствующем масштабе. В результате бесконечного повторения этой процедуры, получается геометрический фрактал.

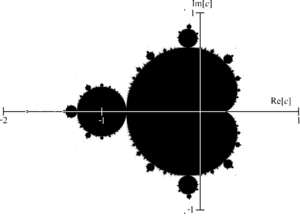
В качестве примера можно привести кривую Коха, изображенную на рисунке 1.1



*Рисунок 1.1*

Второй тип – алгебраические фракталы. Их получают с помощью алгебраических формул. Классическим примером таких фракталов является множество Мандельброта. В основе алгоритма его построения лежит рекуррентное выражение:

где и *c* — комплексные переменные. То есть, это множество таких c, для которых существует такое действительное R, что неравенство < R выполняется при всех натуральных i. Результат моделирования такого множества можно увидеть на рисунке 1.2.



*Рисунок 1.2*

Еще одним известным типом фракталов являются стохастические фракталы. Они в случае, если в итерационном процессе изменять какие-либо его параметры. Двумерные стохастические фракталы используются при моделировании рельефа местности и поверхности моря

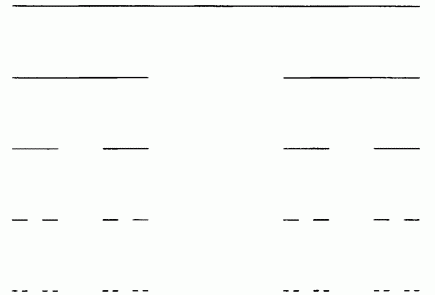
Параграф 1.2

Фрактальная размерность

Как уже было сказано в введении, теоретическая база фракталов была заложена еще задолго до Мандельброта. Одной из первых попыток описать фракталоподобную структура была так называемая «Пыль Кантора» или «Множество Кантора», описанное в 1883 году.

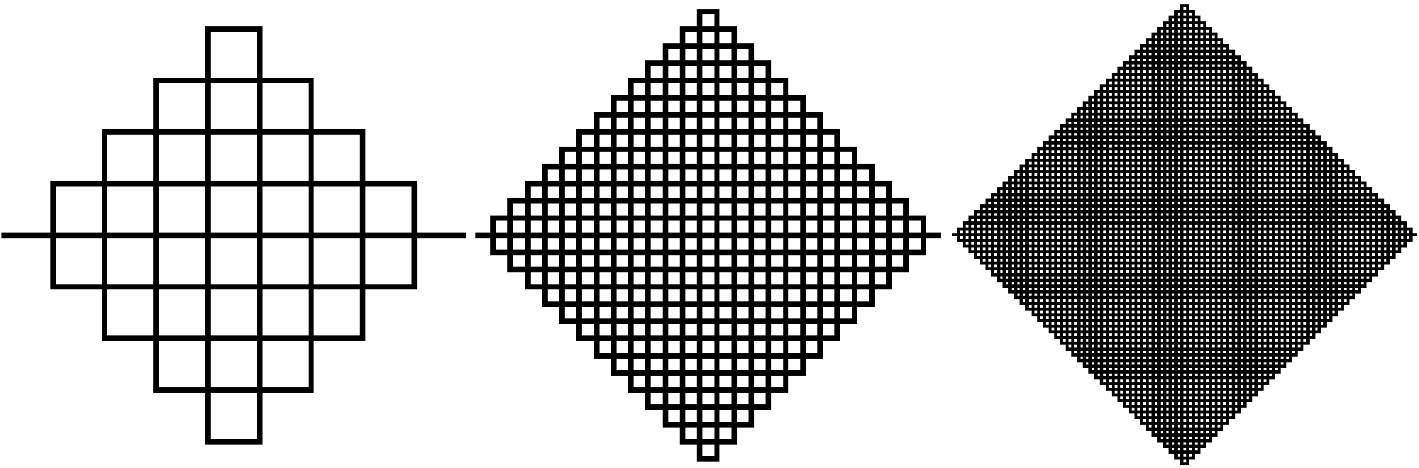
Построение классического множества Кантора начинается с выбрасывания

средней трети единичного отрезка. На следующем и всех остальных шагах мы выкидываем среднюю треть отрезков, получившихся в ходе предыдущего удаления. Таким образом, мы получаем последовательность, изображенную на рисунке 1.3.

  
*Рисунок 1.3*

Также, одной из первых фракталоподобных структур можно считать так называемую кривую Пеано.

Ее построение заключается в том, что отрезок прямой линии единичного размера заменяется на 9 отрезков, длина каждого из которых в 3 раза меньше длины исходного отрезка. После этого такое же правило применяется к каждому из этих девяти отрезков, и т.д. Таким образом получается линия как на рисунке 1.4.

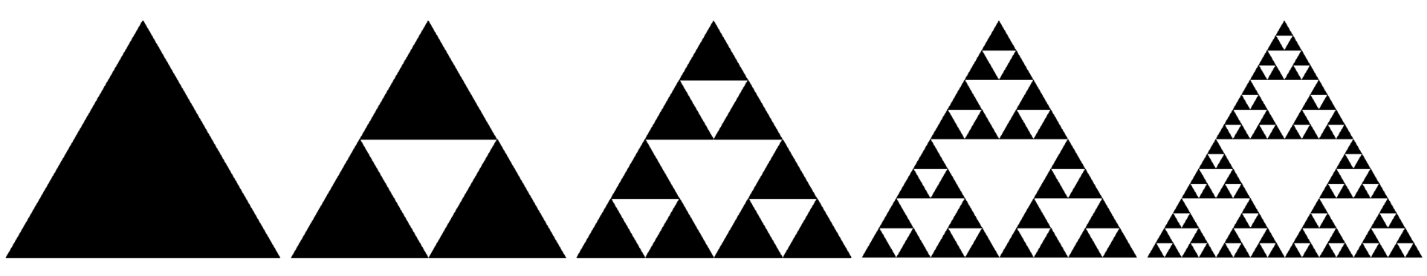


*Рисунок 1.4*

Можно заметить, что Пыль Кантора строится на основании одномерной прямой, но после большого количества итерации начинает состоять из точек, в то время как кривая Пеано при увеличении количества итераций полностью начинает заполнять квадратную область. Становится понятно, что для этих двух геометрических объектов нельзя выделить четкую целую размерность. Это подводит нас к основополагающему понятию фрактальной геометрии – фрактальной размерности.

Важно обозначить, что размерность здесь понимается скорее как зависимость меры фигуры (длины, площади, объема) от масштаба ее рассмотрения. Понятие размерности является центральным понятием, по которому можно охарактеризовать фрактал. Давайте на примере одного из фракталов попробуем вычислить его размерность, а также проанализируем, что она показывает.

Фрактал, чью размерность мы будем изучать – это треугольник Серпинского. Чтобы его получить, нужно взять равносторонний треугольник, провести в нём средние линии и выкинуть центральный из четырех образовавшихся маленьких треугольников. Дальше эти же действия нужно повторить с каждым из оставшихся трех треугольников, и т. д., как на рисунке 1.5.

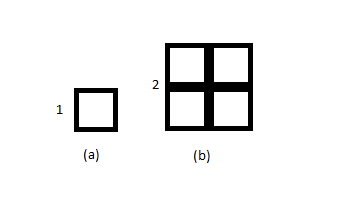


*Рисунок 1.5*

Для того чтобы посчитать размерность используют формулу 1.

Где «d» – размерность, «n» - количество меньших копий фигуры, нужных для того, чтобы получить фигуру со сторонами в «k» раз больше.

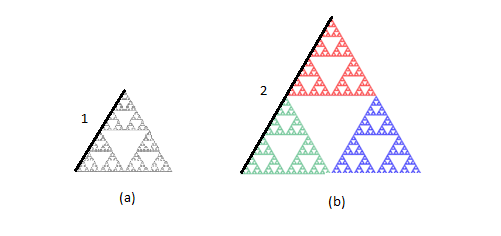
Давайте при помощи этой формулы вычислим размерность для прямоугольников на рисунке 1.6.



*Рисунок 1.6*

Допустим у нас есть квадрат (a) и квадрат (b) увеличенный в 2 раза. Квадрат (b) состоит из 4 квадратов (a). Подставим значения в формулу и вычислим размерность.

Мы получили размерность, равную 2. Что верно, ведь если мы увеличиваем масштаб рассмотрения прямоугольника (увеличивая размер его сторон), то его мера, а именно площадь, увеличивается в 4 раза.

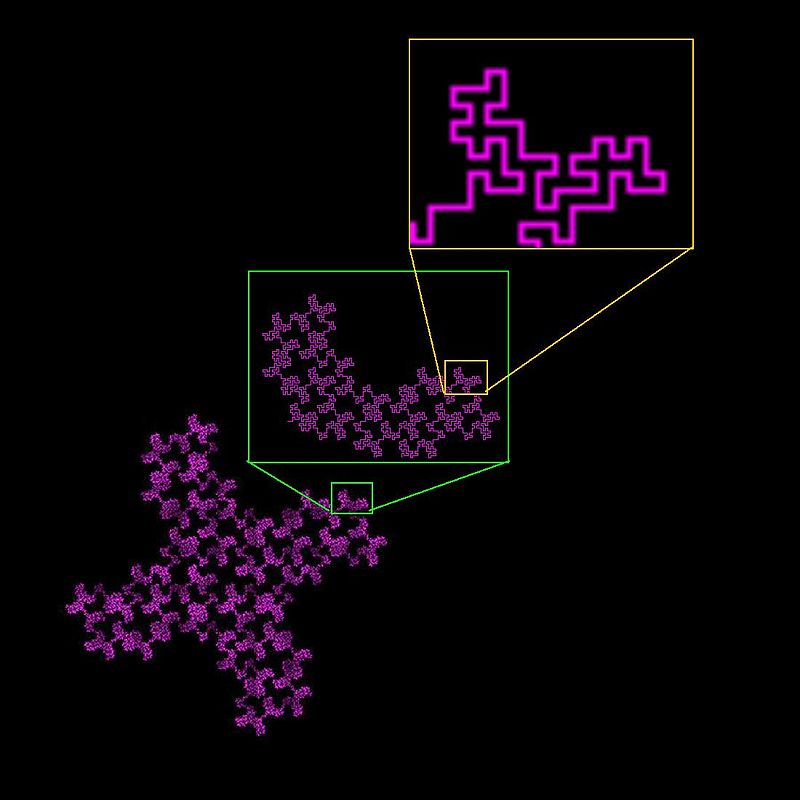


*Рисунок 1.7*

Теперь вернемся к треугольнику Серпинского. Проведем аналогичные операции, как и с квадратом, но с треугольниками на рисунке 1.7. Обозначим длину треугольника (a) за единицу, а затем построим треугольник (b), размером в 2 раза больше. Можно заметить, что треугольник (b) состоит из 3 треугольников (a), что также обосновано построением. Теперь подставим значения в формулу.

Получается, что размерность треугольника Серпинского является дробным числом. Дробная размерность показывает, что мы можем лишь приблизительно описать эту фигуру при помощи линий. Увеличив масштаб в два раза, мы получаем что мера пространства, которое занимает фигура увеличивается в 3 раза. Эта мера не будет являться ни площадью, ни длиной, ведь если попробовать измерить площадь треугольника Серпинского, то обнаружится что она нулевая, а его длина бесконечна.

Очевидно, что фрактальное множество заполняет пространство не так как его заполняет обычное геометрическое множество. В таком случае под размерностью понимают немного другое значение. Фрактальная размерность описывает структуры или множества на основе количественной оценки их «сложности», как коэффициент изменения детали с изменением масштаба. Видно, что на рисунке 1.8 фрактал, в зависимости от масштаба его рассмотрения, имеет самоподобную, но немного отличающуюся структуру, охарактеризованную его размерностью



*Рисунок 1.8*

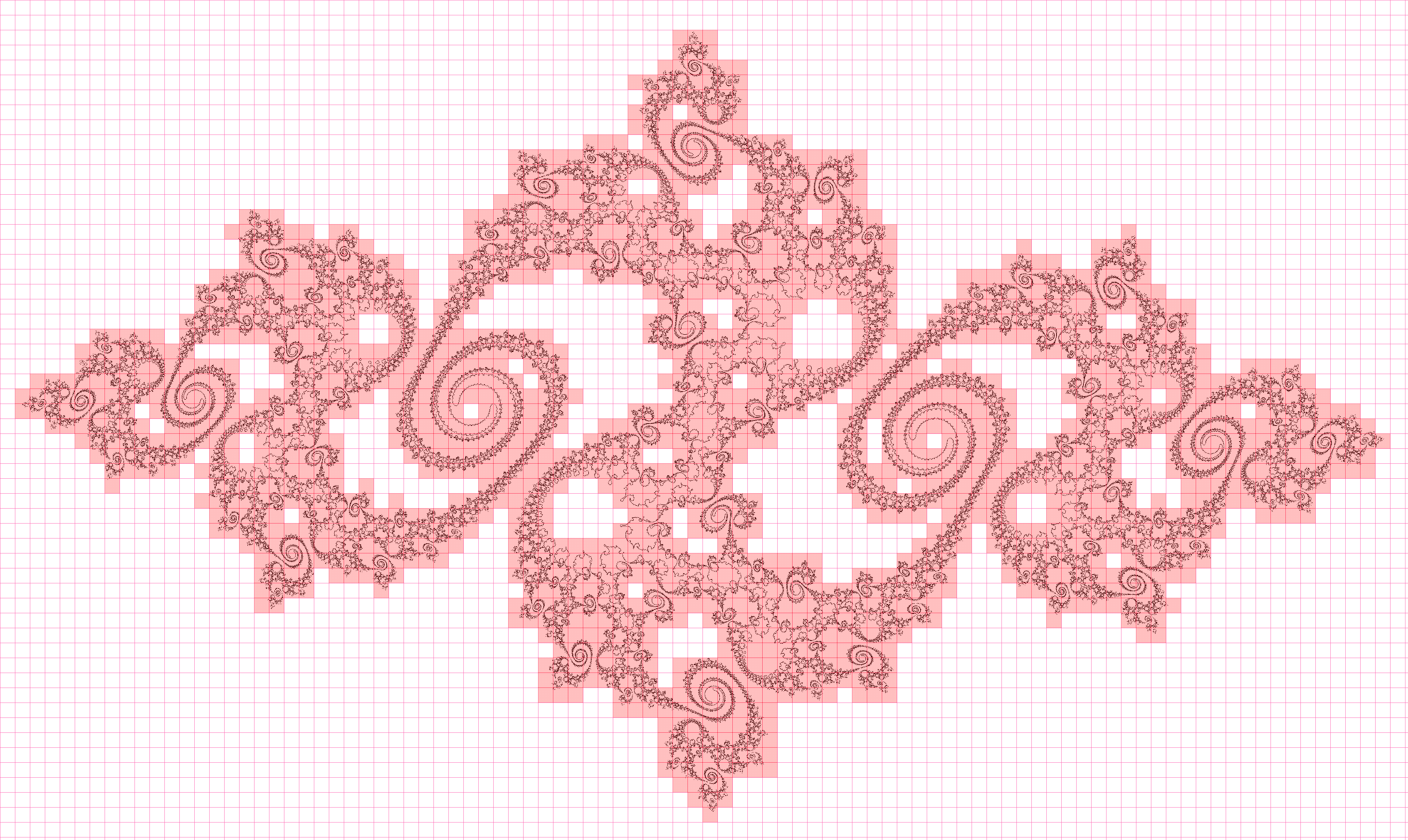
Таким образом фрактальная размерность скорее показывает то, насколько плотно и равномерно фрагменты множества заполняют евклидово пространство.

Параграф 1.3  
Применение фрактальной размерности

Изучение фрактальной размерности заставило ученых немного пересмотреть способ ее вычисления, а также натолкнуло на применение понятия фрактальная размерность в реальной жизни. Оказалось, что фрактальную размерность можно вычислить и у реальных объектов в метрическом пространстве.

Один из способов определения фрактальной размерности множества S в евклидовом пространстве или, в более общем случае, в метрическом пространстве является Размерность Минковского или грубая размерность.

Чтобы вычислить эту размерность для фрактала S, нужно представить, что этот фрактал лежит на равномерно распределенной сетке, такой, как например на рисунке 1.9, и посчитать, сколько ячеек этой сетки требуется, чтобы покрыть множество. Размерность рассчитывается путем наблюдения того, как меняется это число, когда мы делаем сетку более точной, применяя алгоритм подсчета ячеек.



*Рисунок 1.9*

Предположим, что N (ε) - количество ячеек сетки с длиной стороны ε, необходимых для покрытия множества. Тогда размерность в этом методе определяется как:

Такой алгоритм как оказалось можно применить и к природным объектам, посчитав то, какое пространство они занимают плоскости. Одним из самых распространённых примеров является вычисление фрактальной размерности береговой линии Великобритании, рисунок 1.10.

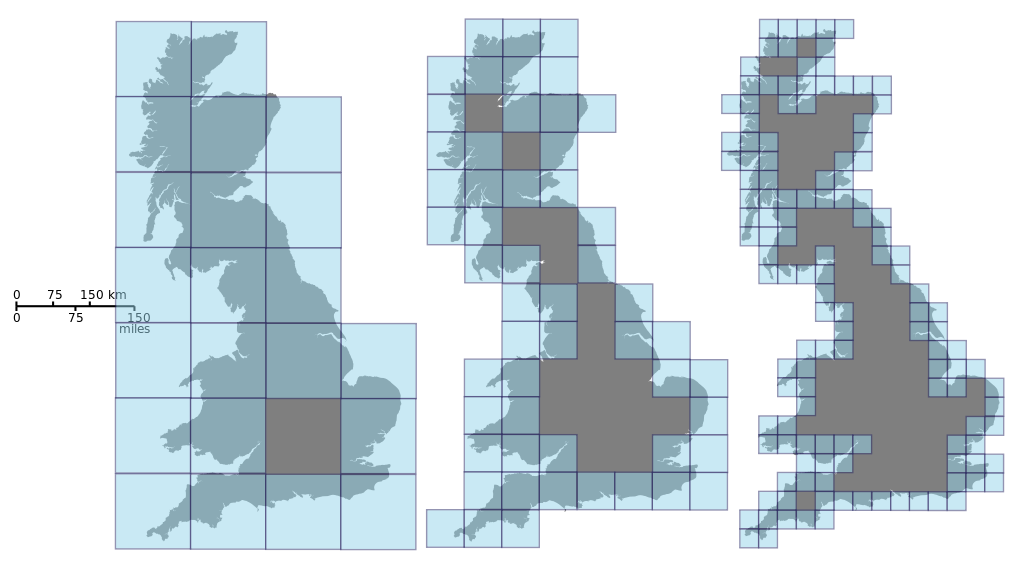


Рисунок 1.10

Для таких объектов как береговая линия показатель фрактальной размерности будет скорее означать коэффициент их неровности, изрезанности и сложности.

Практика вычисления такого коэфициента открывает возможность моделировать структуры подобные природным на компьютере. Процессы рендеринга изображений, отрисовки объектов в кино и видеоиграх завязано на фрактальной геометрии. Пожалуй, это одно из важнейших применений фрактальной размерности и фракталов в целом.

Параграф 1.4

Иные применения фракталов

Одним из применений фракталов в информатике и программировании, помимо отрисовки объектов изображений, является сжатие изображений. В основе этого вида сжатия лежит обнаружение самоподобных участков в изображении. Данный алгоритм известен тем, что в некоторых случаях позволяет получить очень высокие коэффициенты сжатия при приемлемом визуальном качестве для реальных фотографий природных объектов.

В физике фракталы применяются ещё шире. Например, фракталы позволяют точно описывать и предсказывать свойства реальных фракталоподобных тел, имеющих пористую или губчатую структуру, а также различных гелей. Это помогает в создании новых материалов с необычными и полезными свойствами.

Фрактальная геометрия также важна при изучении механики жидкостей. Изучение турбулентности в потоках очень хорошо подстраивается под фракталы. Турбулентные потоки хаотичны и поэтому их сложно точно смоделировать. И здесь помогает переход к фрактальному представлению, что сильно облегчает работу инженерам и физикам, позволяя им лучше понять динамику сложных потоков.

В телекоммуникациях фракталы используются для создания фрактальных антенн. Фрактальные антенны – относительно новый класс электрически малых антенн (ЭМА), принципиально отличающийся своей геометрией от известных решений. По сути, традиционная эволюция антенн базировалась на евклидовой геометрии, оперирующей объектами целочисленной размерности (линия, круг, эллипс, параболоид и т. п.). Фрактальная антенны с удивительно компактным дизайном обеспечивает превосходную широкополосную производительность в маленьком форм-факторе. Достаточно компактны для установки или встраивания в различных местах, фрактальные антенны используются для морских, воздушных транспортных средств, или персональных устройств.

Практическая часть

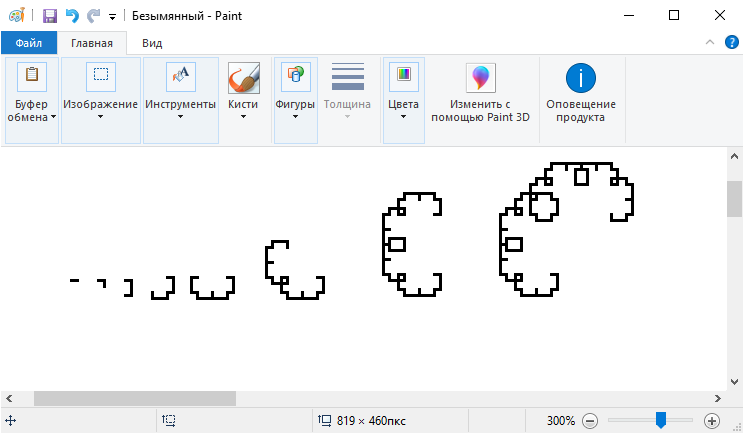
Параграф 2.1

Моделирование фракталов в растровых графических редакторах

Современное ПО позволяет создавать фрактальные объекты даже при помощи примитивных встроенных растровых редакторов изображений, таких, как например Paint.

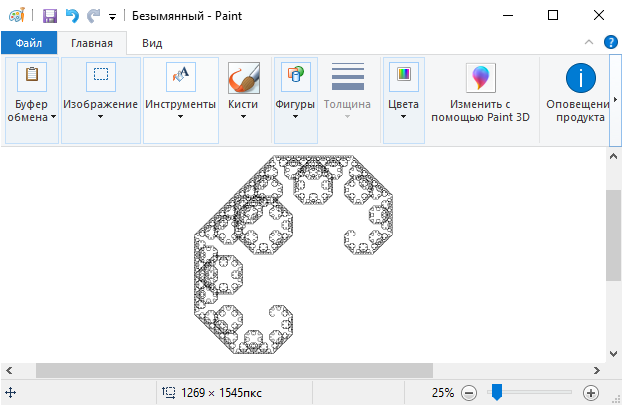
В таких графических редакторах с легкостью можно смоделировать геометрические фракталы, для алгебраических же придется воспользоваться средствами языков программирования.

Начать можно с того, чтобы определить исходный паттерн, по которому будет строиться фрактал. Пускай это будет примитивная линия, состоящая из 3 пикселей, к нижней части которой мы будем присоединять такую же линию, повернутую на 90 градусов. Повторим эту операцию несколько раз, так как это сделано на рисунке 2.1



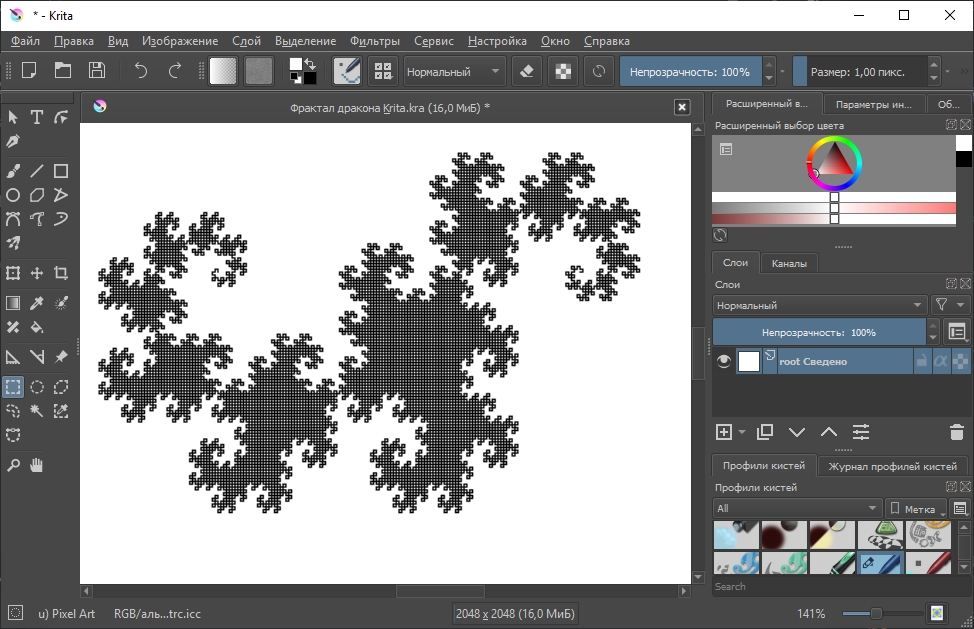
*Рисунок 2.1*

Проделав подобным образом некоторое количество итераций, на рисунке 2.2 получаем фрактал, известный как кривая Леви.



*Рисунок 2.2*

Аналогичным образом строятся фракталы и в других растровых редакторах, таких, как например Adobe Photoshop, Krita и др. В качестве примера на рисунке 2.3 изображен фрактал Дракона, созданный в графическом редакторе Krita.



*Рисунок 2.3*

Параграф 2.2  
Моделирование фракталов при помощи языков программирования

Моделирование фракталов также возможно при помощи современных языков программирования. Они позволяют генерировать как простые геометрические фракталы, так и алгебраические.

Для начала рассмотрим построение примитивных геометрических фракталов, известных как Снежинка Коха и Фрактальное Дерево в языках Python и JavaScript, а затем рассмотрим алгебраический фрактал множество Мандельброта на python

Реализация фракталах в языках программирования завязана на рекурсии. Мы будем реализовывать построение фрактала при помощи простого модуля turtle, способного создавать примитивные графические изображения.

В основе программы будет лежать рекурсивная функция koch, которая будет строить кривую Коха принимающая 5 аргументов:

* t – объект класса Turtle
* iterations – количество итерации при рисовании одной из кривых Коха
* length – длина «под-снежинки»
* shortening\_factor – определяет коэффициент, на который делится длина стороны, когда мы создаем новую «под-снежинку»
* angle – угол под которым появляется новая сторона снежинки

Затем необходимо реализовать само тело функции, основываясь на паттерне построения кривой Коха. Берём отрезок, разделяем на три равные части и заменяем средний интервал равносторонним треугольником без этого сегмента. В результате образуется ломаная, состоящая из четырёх звеньев длины 1/3

def koch(t, iterations, length, shortening\_factor, angle):

    if iterations == 0:

        t.forward(length)

    else:

        iterations = iterations - 1

        length = length / shortening\_factor

        koch(t, iterations, length, shortening\_factor, angle)

        t.left(angle)

        koch(t, iterations, length, shortening\_factor, angle)

        t.right(angle \* 2)

        koch(t, iterations, length, shortening\_factor, angle)

        t.left(angle)

        koch(t, iterations, length, shortening\_factor, angle)

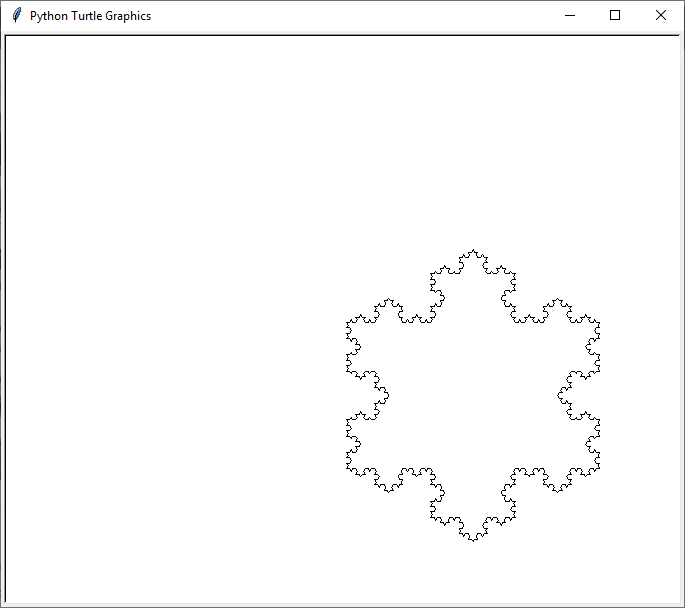
В теле программы вызовем функцию koch 3 раза в цикле, для того чтобы кривые Коха объединились в «Снежинку Коха».

for i in range(3):

    koch(t, 4, 50, 2, 60)

    t.right(120)

Результат работы программы отображен на рисунке 2.4



*Рисунок 2.4*

Возможность моделирования фракталов присутствует практически во всех современных языках программирования. В качестве примера можно привести написанное на JavaScript при помощи фреймвока p5js фрактальное дерево, изображенное на рисунке 2.5.

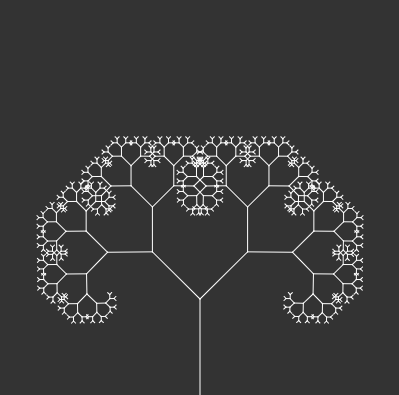


Рисунок 2.5

Теперь рассмотрим построение Множество Мандельброта на языке Python. Строить мы его будем при помощи математической библиотеки numpy и библиотеки для построения графиков matplotlib.

Нам известно, что в основе такого множества лежит рекуррентное выражение:

где и *c* — комплексные переменные. То есть, это множество таких c, для которых существует такое действительное R, что неравенство < R выполняется при всех натуральных i.

Поскольку число c комплексное, у него есть вещественная и мнимая части. Каждое комплексное число задаётся точкой декартовой плоскости: по горизонтальной координате будем откладывать вещественную часть, а по вертикальной — мнимую. Таким образом, множество M является множеством на вещественной плоскости.

Для построения графического изображения множества Мандельброта воспользуемся следующим алгоритмом. Всё множество полностью расположено внутри круга радиуса 2 на плоскости. Поэтому будем считать, что если для точки С последовательность итераций функции после некоторого большого их числа N (скажем, 100) не вышла за пределы этого круга, то точка принадлежит множеству и можно покрасить ее в цвет, соответствующий номеру итерации, на котором её последовательность вышла за пределы круга.

Обозначим некоторые переменные:

|  |  |  |
| --- | --- | --- |
| Переменная(ые) | Тип | Значение |
| pmin, pmax, qmin, qmax | float | Диапазоны, в которых будут меняться значения комплексного числа C |
| ppoints, qpoints | int | Число точек на графике по горизонтали и вертикали |
| max\_iterations | int | Максимальное количество итераций |
| infinity\_border | int | Максимальное количество вычисленных чисел на каждом шаге |

Теперь напишем сам код

import numpy as np

import matplotlib.pyplot as plt

pmin, pmax, qmin, qmax = -2.5, 1.5, -2, 2

ppoints, qpoints = 200, 200

max\_iterations = 300

infinity\_border = 10

def mandelbrot(pmin, pmax, ppoints, qmin, qmax, qpoints,

               max\_iterations=200, infinity\_border=10):

    image = np.zeros((ppoints, qpoints))

    p, q = np.mgrid[pmin:pmax:(ppoints\*1j), qmin:qmax:(qpoints\*1j)]

    c = p + 1j\*q

    z = np.zeros\_like(c)

    for k in range(max\_iterations):

        z = z\*\*2 + c

        mask = (np.abs(z) > infinity\_border) & (image == 0)

        image[mask] = k

        z[mask] = np.nan

    return -image.T

image = mandelbrot(-2.5, 1.5, 1000, -2, 2, 1000)

plt.xticks([])

plt.yticks([])

plt.imshow(image, cmap='flag', interpolation='none')

plt.show()

Результатом его выполнения будет график, представленный на рисунке 2.6

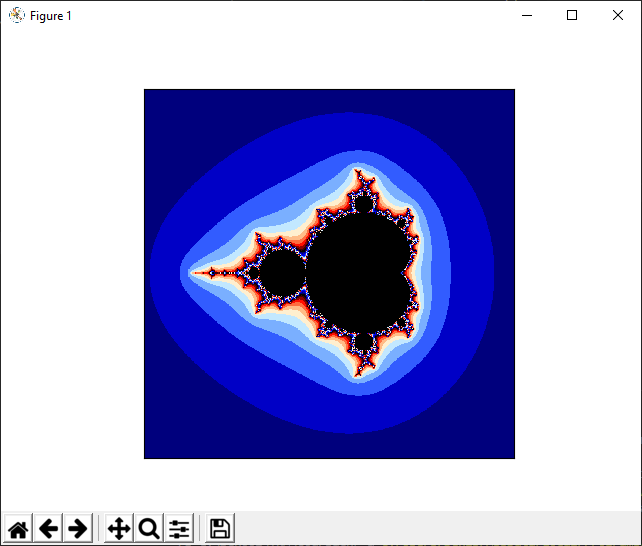


Рисунок 2.6

Также как и с геометрическими фракталами, алгебраические можно представить на большинстве языков программирования. Существует множество онлайн сервисов, на которых можно построить, а также посмотреть приближение множества Мандельброта с заданными параметрами, зачастую такие сервисы реализуются при помощи таких языков как JavaScript и PHP.

Заключение

Фракталы являются уникальными математическими объектами. Они позволяют нам моделировать и изучать с математической точки зрения те объекты, которые ранее считалось невозможно смоделировать. Природа содержит в себе намного больше самоподобия, чем кажется на первый взгляд, и с развитием фрактальной геометрии знания об окружающем нас мире увеличиваются.

Таже важно подчеркнуть, что вычислительная техника и фракталы неразрывно связаны. Без вычислительной техники изучение фракталов практически невозможно, и в то же время моделирование многих объектов на вычислительной технике также не предоставляется возможным без фракталов.

Изучение этих удивительных геометрических фигур как никогда актуально. Многие ученые и инженеры находят фракталам все новые и новые применения в своих разработках.

Источники

Книги

1. Деменок, С. Л. Просто фрактал / С. Л. Деменок. — Санкт-Петербург : Страта, 2019. — 274 с.
2. Ильяшенко Ю. С. И49 Аттракторы и их фрактальная размерность. — М.: МЦНМО, 2005. — 16 с.
3. Х.-О. Пайтген, П.Х. Рихтер, Красота фракталов. Образы комплексных динамических систем, 1993. – 176 с.

Электронные ресурсы

1. Лань: электронно-библиотечная система. — URL: [https://e.lanbook.com/](https://e.lanbook.com/\)
2. Хабр: коллективный блог. – URL: <https://habr.com/ru/post/208368/>