Министерство науки и высшего образования Российской Федерации
федеральное государственное бюджетное образовательное учреждение высшего образования «Иркутский государственный университет» (ФГБОУ ВО «ИГУ»)

Институт математики и информационных технологий Кафедра вычислительной математики и оптимизации

КУРСОВАЯ РАБОТА

по направлению «Прикладная математика и информатика» профиль «Математические методы и информационные технологии»

Построение фракталов на языке Haskell

Студентки 3 курса очного
отделения группы 02321-ДБ
Мезенцевой Виктории Васильевны

Руководитель: к. ф.-м. н., доцент
_____ Черкашин Е. А.

Иркутск – 2022

Содержание

Введение

Haskell – функциональный язык программирования, сильно отличающийся от императивных и смешанных языков разработки. Важной особенность языка является поддержка отложенных вычислений.

Фракталы не новое явление. Они хорошо изучены и имеют обширное применение в жизни. Например, фракталы нашли применение в биологии (моделирование популяций, описание ветвящихся структур), существуют алгоритмы сжатия с помощью фракталов, в компьютерной графике фракталы используются для построения изображение природных объектов – растений, ландшафтов, поверхности морей и т.д. И даже модуляция нашей кровеносной системы или ветви дерева можно построить на фракталах.

Однако в идеи этого явления лежит очень простая логика: бесконечное множество фигур можно получить с помощью всего двух операций – масштабирования и копирования.

Целью данной курсовой работы будет изображение фракталов при помощи языка Haskell.

Теоретическая часть

Фрактал (лат. «fractus» – дроблёный) – самоподобие (копирование) геометрических фигур, где каждый фрагмент дублируется в уменьшающемся масштабе. В природе это явление встречается очень часто.

Можно сказать, что фрактал – это узор, который повторяет сам себя в разных масштабах до бесконечно малого или/и бесконечно большого. Он рождается не просто повторением форм, а скорее повторением процесса, который применяется к форме. Бесконечная цепочка самопостроения.

В природе ярким примером такого узора является капуста сорта «Романеско».

Роль фракталов в машинной графике сегодня достаточно велика. Они приходят на помощь, например, когда требуется, с помощью нескольких коэффициентов, задать линии 4 и поверхности очень сложной формы. С точки зрения машинной графики, фрактальная геометрия незаменима при генерации искусственных облаков, гор, поверхности моря. Фактически найден способ легкого представления сложных неевклидовых объектов, образы которых весьма похожи на природные.

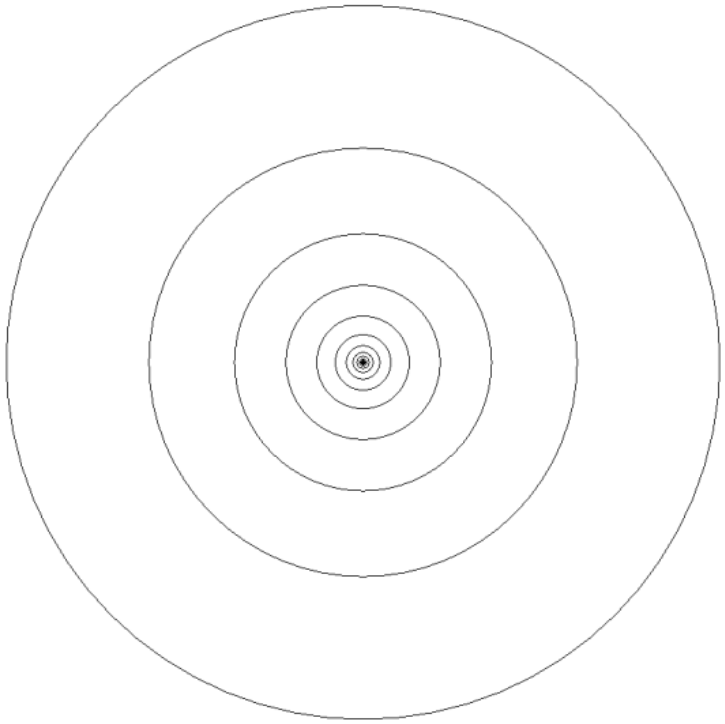
Наиболее полезным использованием фракталов в компьютерной науке является фрактальное сжатие данных. В основе этого вида сжатия лежит тот факт, что реальный мир хорошо описывается фрактальной геометрией.

Практическая часть

Геометрический фрактал

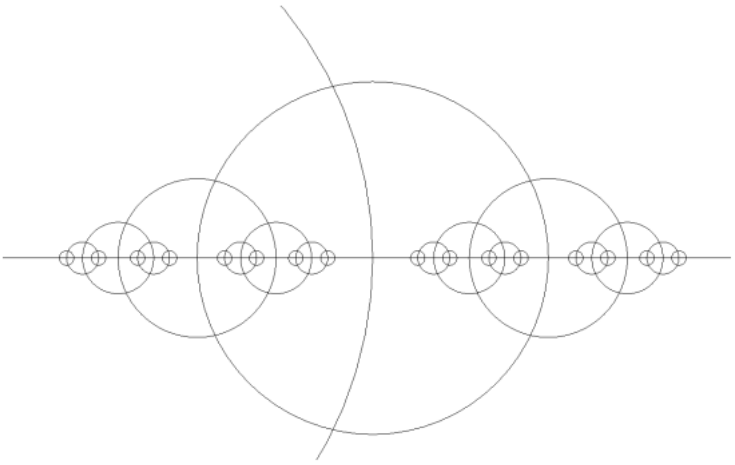
Отрисуем простой фрактал множество кругов, включенных друг в друге. В результате выполнения следующей программы получается серия кругов, каждый из которых нарисован внутри предыдущего круга

```
import Graphics.Gloss
window = InWindow "Fraktal" (800, 800) (20, 20)

circ :: Int -> Picture

circ _ =

   Color black

   $ Scale 100 100

   $ littelCircle 15

littelCircle :: Int -> Picture
littelCircle 0 = Blank
littelCircle n = Pictures [ circle 1, Scale 0.3 0.3 $ littelCircle
(n-1)]
```

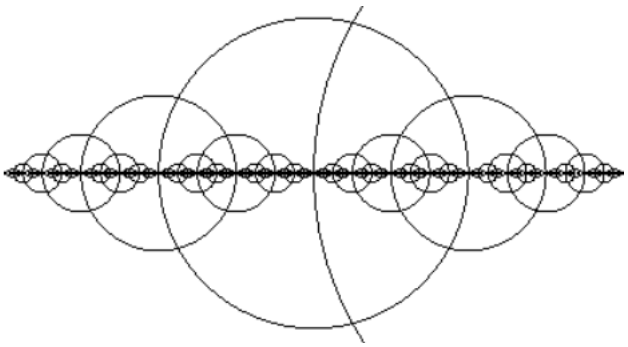


Теперь попробуем усложнить задачу и отрисовать круги со смещениями:

```
import Graphics.Gloss
window = InWindow "Fraktal" (800, 800) (20, 20)

circ :: Int -> Picture
circ 0 = Blank
circ n = Pictures [circ_c, circ_left, circ_right]
   where x = 100
      circ_c = color black $ circle x
      circ_small
         = color black
         $ scale 0.35 0.35
         $ mycirc (n-1)
      circ_right = translate x 0 $ circ_small
      circ_left = translate (-x) 0 $ circ_small

frak = Pictures [line [(-1000,0),(1000,0)], circ 6]
main = display window white frack
```

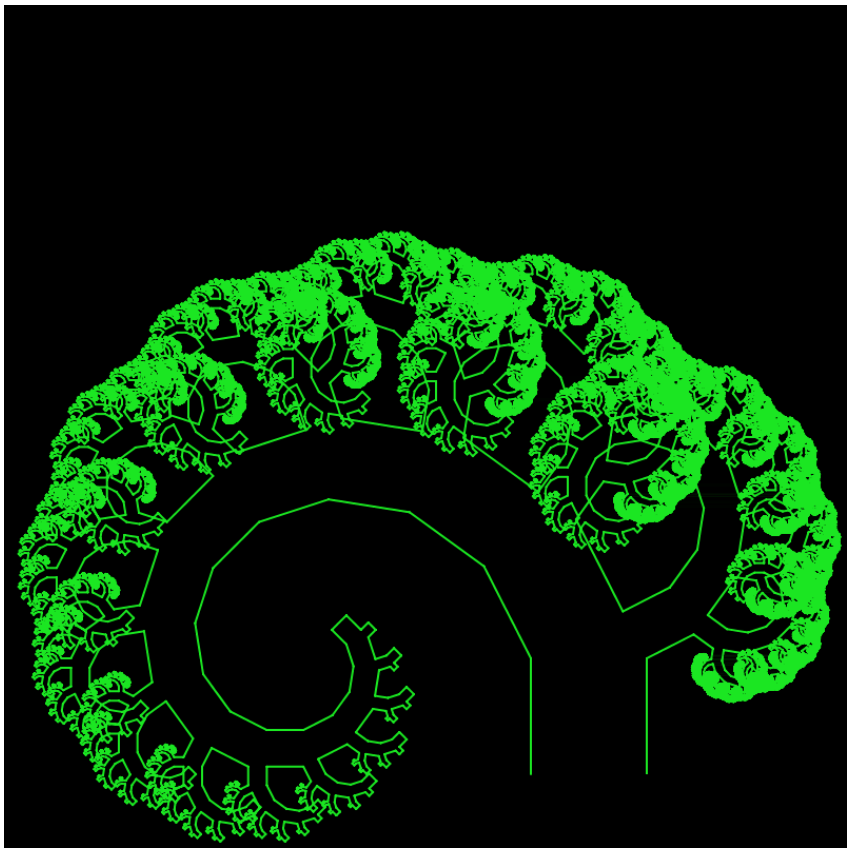


Так как это фрактал при увеличении числа повторений получим более сложную форму фрактала из этих же элементов.



Пифагорейские деревья

Точно так же, как в предыдущем разделе мы сгенерировали геометрические фракталы, используя рекурсию, теперь сгенерируем фигуры, похожие на растения, называемые деревьями Пифагора. Пример дерева Пифагора:



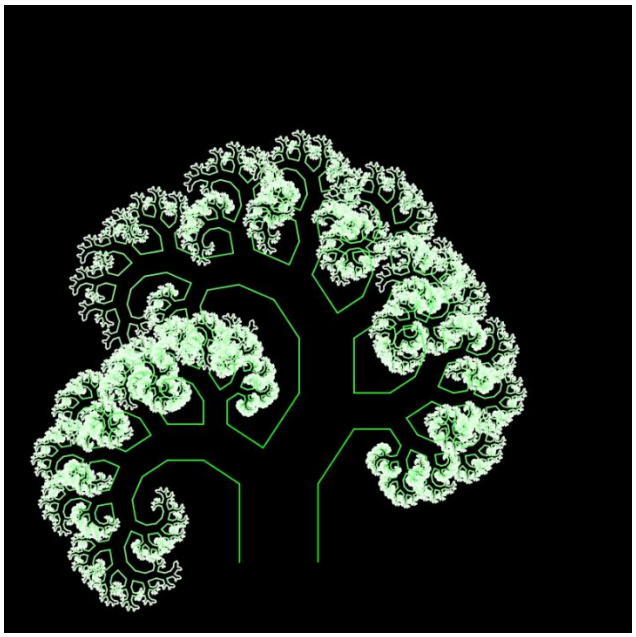
Дерево состоит из последовательности квадратов, наложенных друг на друга. В результате получается удивительно мягкая, органично выглядящая структура.

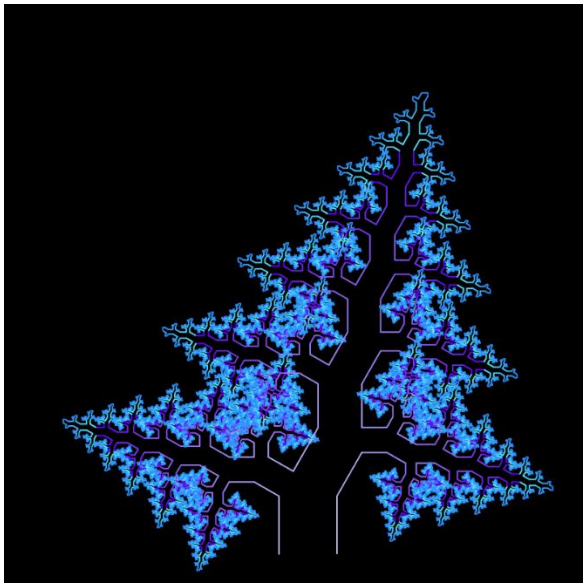
```
import Graphics.Gloss
mywindow = InWindow "Tree" (600, 600) (20, 20)

fractalTree :: Float -> Int -> Line -> Path
fractalTree factor n line = fractalTree' n line
  where
    fractalTree' 0 line = []
    fractalTree' n line
      = [p1, p4] ++ fractalTree' (n-1) (p4, p5) ++
                    fractalTree' (n-1) (p5, p3) ++
        [p3, p2]
      where
        factor = if (n 'mod' 5 == 0)
          then factor
```

```
    else (1 - factor)
flipLine (pS, pE) = (pE, pS)

[p1,p2,p3,p4,_] = polygon 4 line
(_, p5)         = rotateLine (factor * pi) $
                  flipLine $
                     scaleLine 0.5 $ (p3,p4)

let myLine = ((470,800) , (3,800))
drawPicture 2 [(green, fractalTree 0.45 13 myLine)]
```

Результат:

При изменении цветов и параметров можно получить различные варианты деревьев Пифагора:





Заключение

В ходе выполнения курсовой работы был рассмотрен инструментарий языка программирования Haskell для визуализации фракталов. Haskell предоставляет удобные условия для реализации подобных задач.

В курсовой работе продемонстрированы возможности Haskell в визуализации изображений с помощью сторонней библиотеки Gloss, а также рекурсивные вызовы функций, выступающие основным инструментом работы.

Также подобное упражнение отлично и наглядно показывает способности функциональных языков к реализации рекурсивных структур и просто решении задач через создание рекурсивных функций.

Список литературы

1. Г.М. Сергиевский, Н.Г. Волченков. Функциональное и логическое программирование. – М.: Академия, 2010. – 320 с.: ил. ISBN: 978-5-7695-6433-8
2. Уилл Курт. Программируй на Haskell / пер. с англ. Я. О. Касюлевича, А. А. Романовского и С. Д. Степаненко; под ред. В. Н. Брагилевского. – М.: ДМК Пресс, 2019. — 648 с.: ил. ISBN 978-5-97060-694-0
3. Gloss [Электронный ресурс] URL: http://gloss.ouroborus.net
4. Как устроены фракталы: бесконечность и красота математики [Электронный ресурс] URL: https://www.techinsider.ru/science/8906-krasota-povtora-fraktaly/
5. Graphics.Gloss — hackage.haskell.org [Электронный ресурс] URL: https://hackage.haskell.org/package/gloss-1.13.2.2/docs/GraphicsGloss.html