Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ПРАКТИЧЕСКАЯ РАБОТА №2**

**«ФРАКТАЛЫ И ФРАКТАЛЬНАЯ ГРАФИКА»**

Выполнил: студент учебной группы

ИСПк-203-52-00

Леушина Анна Станиславовна

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2024

1. Что такое фрактал? Какие виды бывают?

**Фрактал** (fractus – дробленный, сломанный, разбитый) – это сложная, бесконечно самоподобная геометрическая фигура, каждый фрагмент которой повторяется при уменьшении масштаба.

Виды фракталов:

* **Геометрические фракталы:** эти фракталы обладают геометрическими свойствами самоподобия на различных масштабах. Они могут быть построены с использованием простых геометрических операций, таких как деление отрезка, поворот и масштабирование. Примеры геометрических фракталов включают фрактальную кривую Коха и снежинку Коха.
* **Алгебраические фракталы:** эти фракталы определяются с использованием алгебраических уравнений или рекурсивных формул. Они могут быть построены с использованием математических операций, таких как возведение в степень, сложение и умножение. Примеры алгебраических фракталов включают множество Мандельброта и фрактальное дерево.
* **Стохастические фракталы:** эти фракталы характеризуются случайными или стохастическими процессами, которые используются для их создания. Они могут иметь случайные элементы или изменяющиеся параметры, что приводит к разнообразию форм и структур. Примеры стохастических фракталов включают фрактальный броуновский мост и стохастическую лесенку.

1. Что такое ломанная-генератор?

**Ломанная-генератор** – это алгоритм, который генерирует ломаную линию с использованием заданных правил. Ломанная линия представляет собой последовательность отрезков, соединяющих точки, называемые вершинами.

1. Опишите фрактал из своего варианта ДКР.

**Фрактал снежинка Коха** – это геометрический фрактал, который строится путем замены каждого отрезка на определенную группу отрезков, таким образом, чтобы создавалась сложная структура, напоминающая снежинку. Фрактал Коха состоит из главного треугольника, каждая сторона которого заменяется на равносторонний треугольник.

1. Укажите ломанную-генератор и опишите алгоритм построения фрактала в зависимости от глубины.

**Алгоритм построения:**

* Задать начальную длину стороны треугольника (или отрезка) и начальную глубину рекурсии.
* Разделить сторону на три равные части.
* Построить четыре отрезка: два отрезка с длиной 1/3 и углом 60 градусов к исходной стороне, один отрезок с длиной 1/3 и углом -60 градусов, и один отрезок между концами этих трех отрезков.
* Уменьшить глубину рекурсии на 1 и повторить шаги 2-4 для каждого из трех новых отрезков.
* Продолжать рекурсивно уменьшать глубину и строить новые отрезки до тех пор, пока глубина не станет равной нулю.

**Ломанная генератор:**

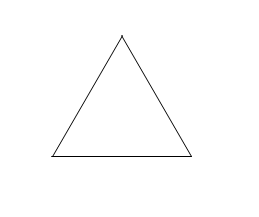


Рисунок 1 – Глубина 0

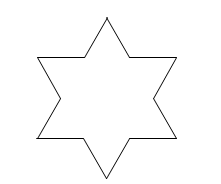


Рисунок 2 – Глубина 1

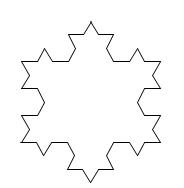


Рисунок 3 – Глубина 2

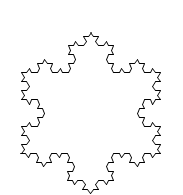


Рисунок 4 – Глубина 3

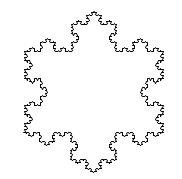


Рисунок 5 – Глубина 4

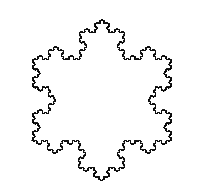


Рисунок 6 – Глубина 5

1. Запишите рекурсивную подпрограмму для построения фрактала, в ней укажите рекурсивную триаду.

**Рекурсивная подпрограмма для построения фрактала:**

procedure Draw(x, y, l, u: Real; t: Integer);

procedure Draw2(var x, y: Real; l, u: Real; t: Integer);

begin

Draw(x, y, l, u, t);

x := x + l \* cos(u);

y := y - l \* sin(u);

end;

begin

if t > 0 then

begin

l := l / 3;

Draw2(x, y, l, u, t - 1);

Draw2(x, y, l, u + pi / 3, t - 1);

Draw2(x, y, l, u - pi / 3, t - 1);

Draw2(x, y, l, u, t - 1);

end

else

Line(Round(x), Round(y), Round(x + cos(u) \* l), Round(y - sin(u) \* l))

end;

**Рекурсивная триада:**

procedure Draw(x, y, l, u: Real; t: Integer); //параметризация

begin

if t > 0 then //база рекурсии

begin

l := l / 3;

Draw2(x, y, l, u, t - 1); //Декомпозиция

Draw2(x, y, l, u + pi / 3, t - 1); // Декомпозиция

Draw2(x, y, l, u - pi / 3, t - 1); // Декомпозиция

Draw2(x, y, l, u, t - 1); // Декомпозиция

end

else

Line(Round(x), Round(y), Round(x + cos(u) \* l), Round(y - sin(u) \* l))

end;