Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №7**

**«ИССЛЕДОВАНИЕ ФРАКТАЛОВ»**

**ПО МДК 05.02 РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ**

Выполнил:

Домрачев Артём Андреевич

ИСПк-202-52-00

Колледж ВятГУ

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

1. **Цель контрольной работы:**

Получение навыков реализации алгоритмов с рекурсивными вычислениями, знакомство с фракталами.

1. **Формулировка задания:**

Вариант: 17

Задание:

1. Написать программу для визуализации фрактала "Кривая Хартера-Хейтуэя".

2. Предусмотреть возможности масштабирования, изменения глубины прорисовки и перемещения полученной фигуры.

3. Построение множества ломанных, образующих фрактал, должно осуществляется в отдельном модуле.

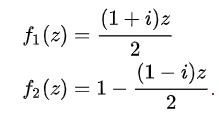
1. **Описание алгоритма:**

Фрактал может быть записан как [L-система](https://ru.wikipedia.org/wiki/L-%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) с параметрами:

* угол равен 90° или pi/2
* начальная строка — *FX*
* правила преобразования строк:



Кроме того, фрактал может быть описан системой итерируемых функций на комплексной плоскости:



Берём отрезок, сгибаем его пополам. Затем многократно повторяем итерацию. Если после этого снова разогнуть получившуюся (сложенную) линию так, чтобы все углы были равны 90°, мы получим драконову ломаную.



Рисунок 1 – Построение кривой дракона



Рисунок 1.1 – Построение кривой дракона



Рисунок 1.2 – Построение кривой дракона

1. **Код программы:**

**Uses** GraphABC;

**var** (m,wx,wy,g) := (100, 300,300,10);

**Procedure** ris(x1,y1,x2,y2,k:integer);

**begin**

**If** k>0 **then begin**

**var** xn:=(x1+x2) **div** 2 +(y2-y1) **div** 2;

**var** yn:=(y1+y2) **div** 2 -(x2-x1) **div** 2;

ris(x1,y1,xn,yn,k-1);

ris(x2,y2,xn,yn,k-1)

**end**

**else** line(x1,y1,x2,y2)

**end**;

**Procedure** KeyDown(k: integer);

**begin**

**case** K **of**

VK\_Down: wy -= 10;

VK\_Up: wy += 10;

VK\_Left: wx += 10;

VK\_Right: wx -= 10;

VK\_Escape: halt(1); //выход

VK\_A: **if** m < 500 **then** m+=10;

VK\_Z: **if** m > 20 **then** m-=10;

VK\_S: **if** g < 15 **then** g+=1;

VK\_X: **if** g > 2 **then** g -=1;

**end**;

Window.Clear;

ris(wx-m,wy,wx+m,wy,g);

Redraw;

**end**;

**Begin**

SetWindowCaption('Фракталы: Кривая Дракона');

SetWindowSize(700,512);

LockDrawing;

KeyDown(0);

OnkeyDown += KeyDown;

**End**.

1. **Результат выполнения программы:**

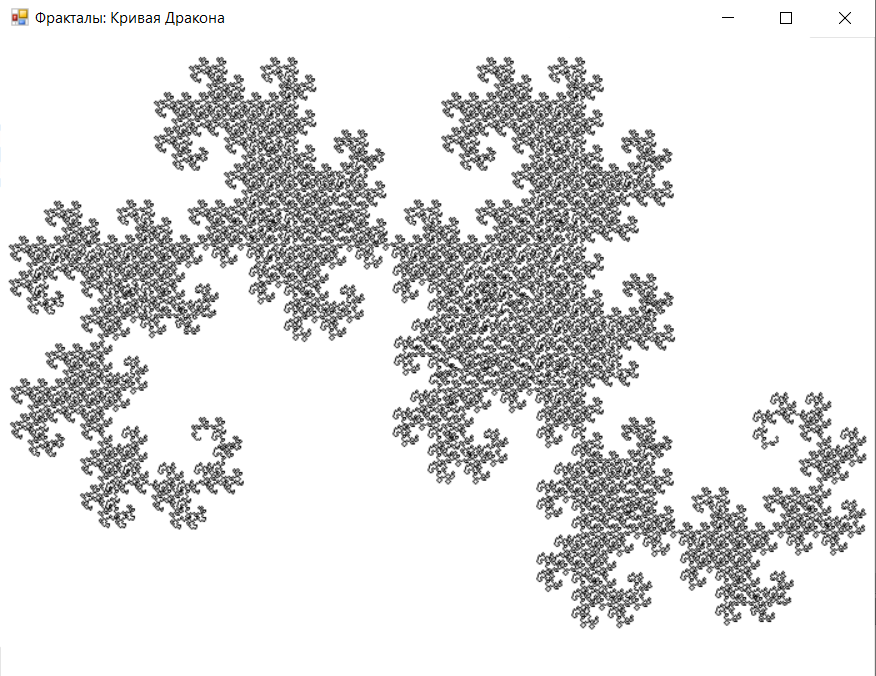


Рисунок 2 – Кривая Хартера-Хейтуэя

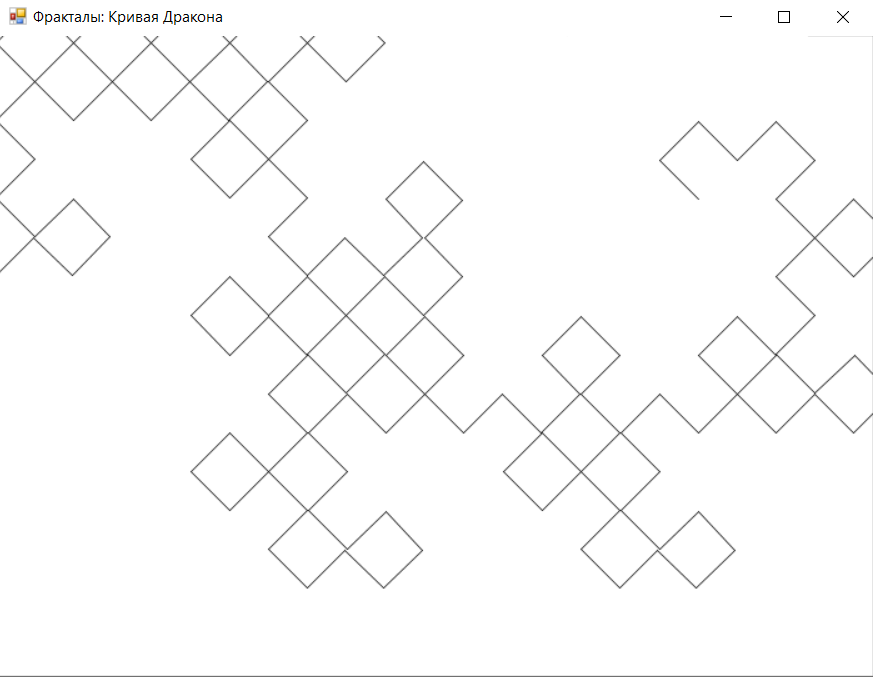


Рисунок 3 – Приближение

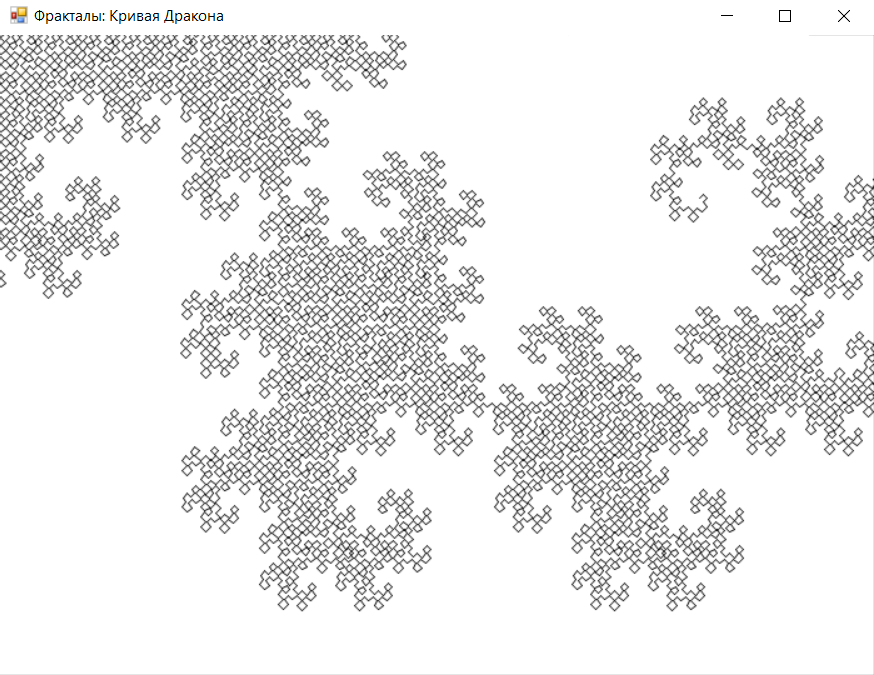


Рисунок 4 – Движение

1. **Вывод:**

В процессе реализации алгоритмов с использованием рекурсивных вычислений мы приобрели базовые навыки и знания. Мы изучили и поняли, как создать кривую, известную как "Кривая Хартера-Хейтуэя". Также мы ознакомились с понятием рекурсии и способами ее формирования. Рекурсия представляет собой определение объекта путем ссылки на самого себя. Мы научились делать приближение и движение в контексте рекурсивных алгоритмов.