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**

**«ИСЛЕДОВАНИЕ ФРАКТАЛА»**

**ПО МДК 05.02 РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ**

Выполнил: студент учебной группы

ИСПк-204-52-00

Воробьев Алексей Владимирович

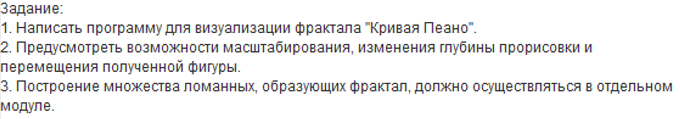
Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

1. Цель работы: получение навыков реализации алгоритмов с рекурсивными вычислениями, знакомство с фракталами



1. Описание алгоритма

Программа начинается с входных данных (начальное кол-во итераций, начальные координаты и расстояние между линиями). После идет функция per которая отвечает за очистку окна после каждого нажатия на кнопки управления, за кнопку которую нажал пользователь и вход в цикл рисования фрактала. Далее идут функции KU, KD, KR, KL которые нужны для вызова самих себя и LineUP, LineDOWN, LineRIGHT, LineLEFT которые рисуют линии и описаны в библиотеке fractal.

1. Код программы

**uses** GraphABC, fractal;

**var**

x0, y0, x00, y00, step, iter: integer;

**function** KU(i:integer):integer; **forward**;

**function** KD(i:integer):integer; **forward**;

**function** KR(i:integer):integer; **forward**;

**function** KL(i:integer):integer; **forward**;

**function** per: integer; **forward**;

**function** KU( i: integer): integer;

**begin**

**if** i > 0 **then**

**begin**

KR(i - 1);

LineUP(x0, y0, step);

KU(i - 1);

LineRIGHT(x0, y0, step);

KU(i - 1);

LineDOWN(x0, y0, step);

KL(i - 1);

**end**;

**end**;

**function** KD( i: integer): integer;

**begin**

**if** i > 0 **then**

**begin**

KL(i - 1);

LineDOWN(x0, y0, step);

KD(i - 1);

LineLEFT(x0, y0, step);

KD(i - 1);

LineUP(x0, y0, step);

KR(i - 1);

**end**;

**end**;

**function** KL( i: integer): integer;

**begin**

**if** i > 0 **then**

**begin**

KD(i - 1);

LineLEFT(x0, y0, step);

KL(i - 1);

LineDOWN(x0, y0, step);

KL(i - 1);

LineRIGHT(x0, y0, step);

KU(i - 1);

**end**;

**end**;

**function** KR( i: integer): integer;

**begin**

**if** i > 0 **then**

**begin**

KU(i - 1);

LineRIGHT(x0, y0, step);

KR(i - 1);

LineUP(x0, y0, step);

KR(i - 1);

LineLEFT(x0, y0, step);

KD(i - 1);

**end**;

**end**;

**procedure** control(key:integer);

**begin**

**case** key **of**

VK\_Up : **begin** y00:=y00-10; per; **end**;

VK\_Down : **begin** y00:=y00+10; per; **end**;

VK\_Left : **begin** x00:=x00-10; per; **end**;

VK\_Right : **begin** x00:=x00+10; per; **end**;

VK\_q : **begin** step:=step+2; per; **end**;

VK\_w : **begin if** step > 0 **then** step:=step-2; per; **end**;

VK\_z : **begin if** iter<10 **then** iter:=iter+1; per; **end**;

VK\_x : **begin if** iter>0 **then** iter:=iter-1; per; **end**;

**end**;

**end**;

**function** per: integer;

**begin**

MaximizeWindow;

clearwindow;

x0:=x00;

y0:=y00;

LockDrawing;

KU(iter);

redraw;

onKeyDown:=control;

**end**;

**begin**

iter:=1;

x00:=20;

y00:=1000;

step:=100;

per;

**end**.

**Модуль фрактала**

**unit** fractal;

**interface**

**uses** GraphABC;

**procedure** LineUP(**var** x0, y0, step: integer);

**procedure** LineDOWN(**var** x0, y0, step: integer);

**procedure** LineRIGHT(**var** x0, y0, step: integer);

**procedure** LineLEFT(**var** x0, y0, step: integer);

**implementation**

**procedure** LineUP(**var** x0, y0, step: integer);

**begin**

line(x0, y0, x0, y0 - step);

y0 := y0 - step;

**end**;

**procedure** LineDOWN(**var** x0, y0, step: integer);

**begin**

Line(x0, y0, x0, y0 + step);

y0 := y0 + step;

**end**;

**procedure** LineRIGHT(**var** x0, y0, step: integer);

**begin**

Line(x0, y0, x0 + step, y0);

x0 := x0 + step;

**end**;

**procedure** LineLEFT(**var** x0, y0, step: integer);

**begin**

Line(x0, y0, x0 - step, y0);

x0 := x0 - step;

**end**;

**end**.

5. Результат выполнения программы

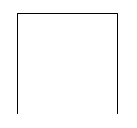


Рисунок 1 – Результаты выполнение программы

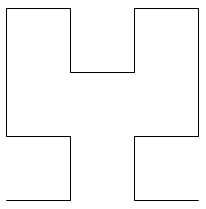


Рисунок 2 – Результаты выполнение программы

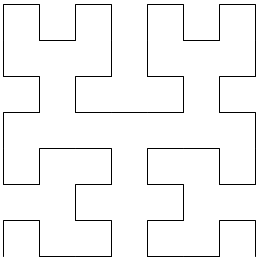


Рисунок 3 – Результаты выполнение программы

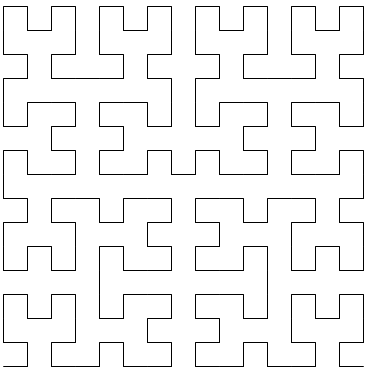


Рисунок 4 – Результаты выполнение программы

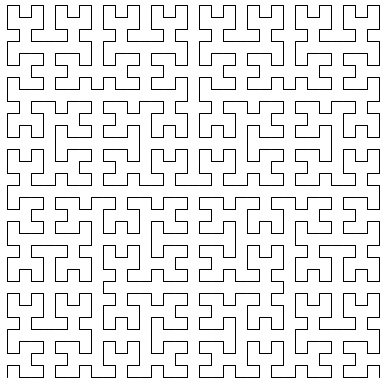


Рисунок 5 – Результаты выполнение программы

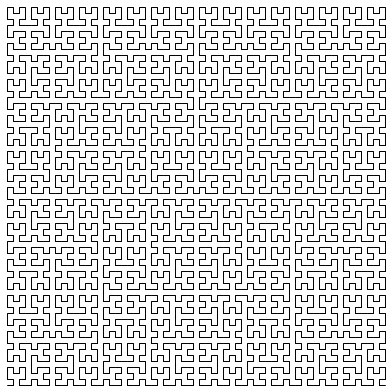


Рисунок 6 – Результаты выполнение программы

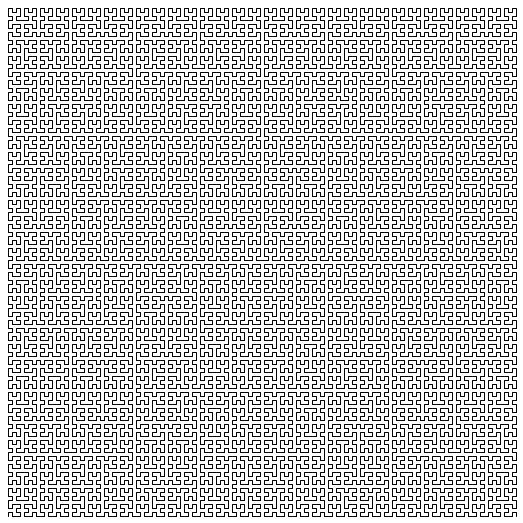


Рисунок 7 – Результаты выполнение программы

6. Вывод

В этой домашней контрольной работе использовалась косвенная рекурсия и была изучена кривая Пеано.

Программа начинается с того что вызывает функцию, которая меняет масштаб, количество итераций фрактала, перемещает фигуру, и вызывать функцию которая будет рисовать фигуру.

В ходе разработки кода встретились проблемы с написанием правильной последовательности рисования линий во фрактале и созданием отдельной библиотеки(модуля) в программе.