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»**

**«МДК05.02 Разработка кода информационных систем»**

Выполнил: студент учебной группы

ИСПк-205-52-00

Лусников Владимир Михайлович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2025

Цель работы

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

Задание:

Формулировка задания (с вариантом)

Вариант 15

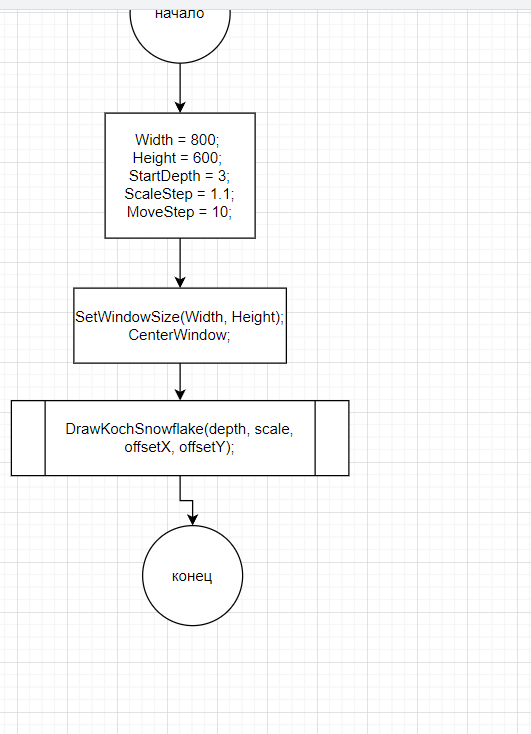
﻿﻿ Задание:

1. Написать программу для визуализации фрактала "Снежинка Коха".

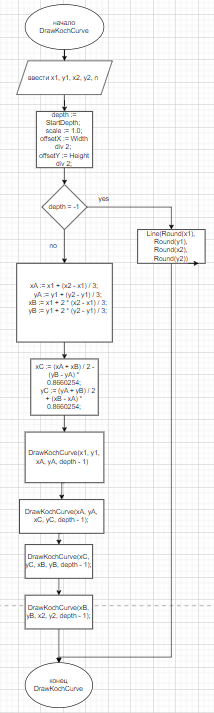
2. Предусмотреть возможности масштабирования, изменения глубины прорисовки и перемещения полученной фигуры.

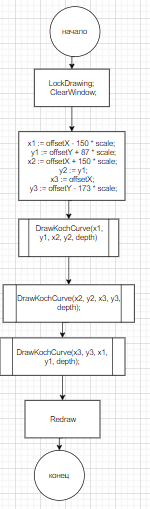
3. Построение множества ломанных, образующих фрактал, должно осуществляться в отдельном модуле.

**Схема алгоритма**

****

****

****

****

Код для программы в Pascal.

**Задание 1**

**Главный код**

**program** Koxa;

**uses** GraphABC, Koxa1;

**const** Width = 800; Height = 600; StartDepth = 3; ScaleStep = 1.1; MoveStep = 10;

**var** depth: Integer; scale: Real; offsetX, offsetY: Integer;

**procedure** KeyDown(Key: Integer);

**begin**

**case** Key **of**

VK\_Z: Inc(depth);

VK\_X: **if** depth > 0 **then** Dec(depth);

VK\_Q: scale := scale \* ScaleStep;

VK\_E: scale := scale / ScaleStep;

VK\_W: Dec(offsetY, MoveStep);

VK\_S: Inc(offsetY, MoveStep);

VK\_A: Dec(offsetX, MoveStep);

VK\_D: Inc(offsetX, MoveStep);

**end**;

DrawKochSnowflake(depth, scale, offsetX, offsetY);

**end**;

**begin**

SetWindowSize(Width, Height); CenterWindow;

depth := StartDepth; scale := 1.0; offsetX := Width **div** 2; offsetY := Height **div** 2;

OnKeyDown := KeyDown;

DrawKochSnowflake(depth, scale, offsetX, offsetY);

**end**.

Модуль

**unit** koxa1;

**interface**

**uses** GraphABC;

**procedure** DrawKochSnowflake(depth: Integer; scale: Real; offsetX, offsetY: Integer);

**implementation**

**procedure** DrawKochCurve(x1, y1, x2, y2: Real; depth: Integer);

//декомпозиция

**var** xA, yA, xB, yB, xC, yC: Real;

**begin**

//база рекурсии

**if** depth = -1 **then** Line(Round(x1), Round(y1), Round(x2), Round(y2)) **else**

**begin**

xA := x1 + (x2 - x1) / 3;

yA := y1 + (y2 - y1) / 3;

xB := x1 + 2 \* (x2 - x1) / 3;

yB := y1 + 2 \* (y2 - y1) / 3;

xC := (xA + xB) / 2 - (yB - yA) \* 0.8660254;

yC := (yA + yB) / 2 + (xB - xA) \* 0.8660254;

//рекурсивные вызовы

DrawKochCurve(x1, y1, xA, yA, depth - 1);

DrawKochCurve(xA, yA, xC, yC, depth - 1);

DrawKochCurve(xC, yC, xB, yB, depth - 1);

DrawKochCurve(xB, yB, x2, y2, depth - 1);

**end**;

**end**;

**procedure** DrawKochSnowflake(depth: Integer; scale: Real; offsetX, offsetY: Integer);

**var** x1, y1, x2, y2, x3, y3: Real;

**begin**

LockDrawing;

ClearWindow;

x1 := offsetX - 150 \* scale;

y1 := offsetY + 87 \* scale;

x2 := offsetX + 150 \* scale;

y2 := y1;

x3 := offsetX;

y3 := offsetY - 173 \* scale;

DrawKochCurve(x1, y1, x2, y2, depth);

DrawKochCurve(x2, y2, x3, y3, depth);

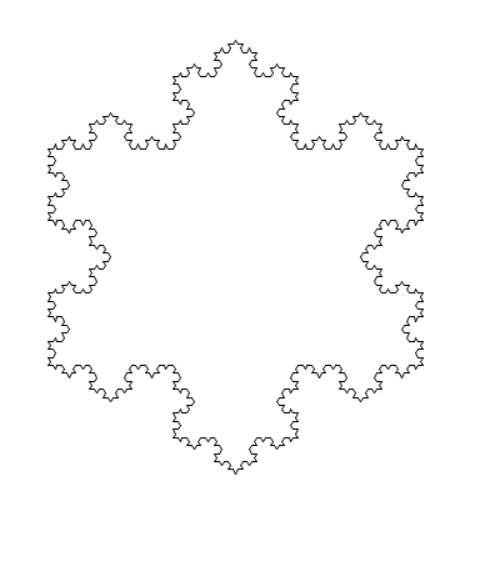
DrawKochCurve(x3, y3, x1, y1, depth);

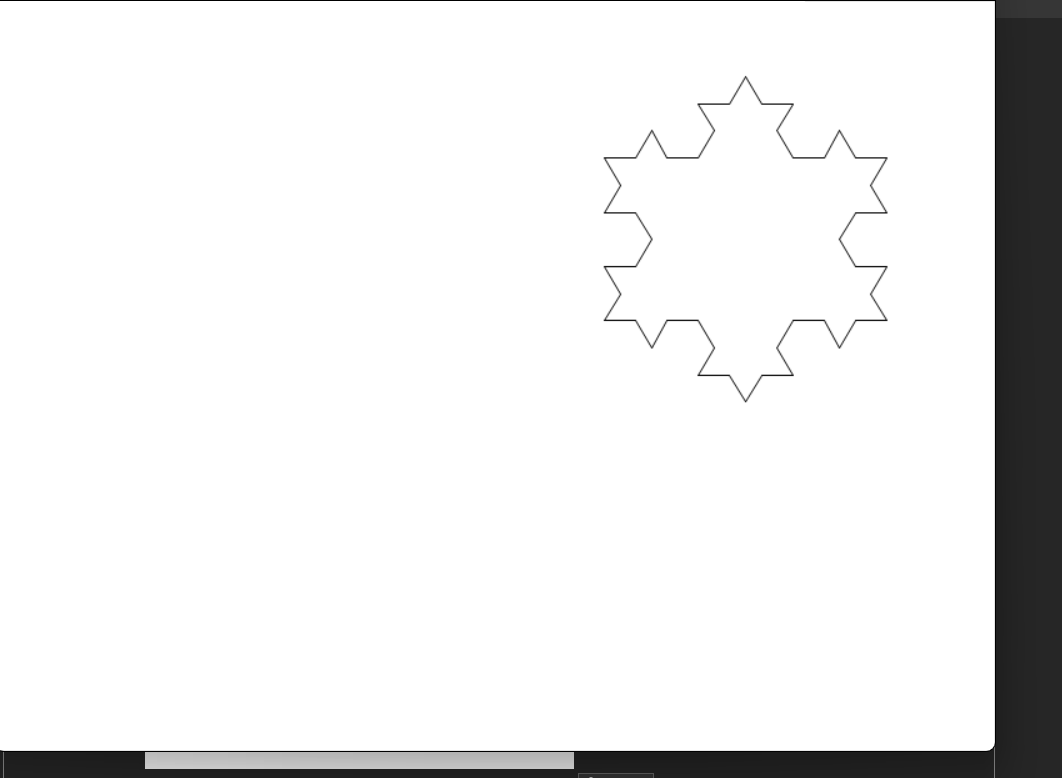
Redraw;

**end**;

**end**.

Результат программы





Вывод:

С помощью домашней контрольной работы №7 я научился строит фрактал "Снежинка Коха" с возможностью масштабирования, изменения глубины прорисовки и перемещения, разделяя построение фрактала и визуализацию на отдельные модули.