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: студент учебной группы

ИСПк-101-51-00

Сунцов Александр Андреевич

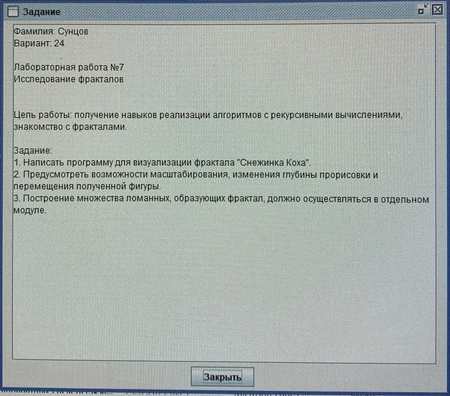
Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

1. **Цель работы:** Получение навыков реализации алгоритмов с рекурсивными вычислениями, знакомство с фракталами
2. **Скриншот задания (с вариантом):**

****

1. **Описание алгоритма**

Строится равносторонний треугольник (0 итерация). Далее каждая сторона треугольника делится на три одинаковые части, после чего рисуется равносторонний треугольник, у которого основание – средний сегмент отрезка, и основание построенного треугольника на отрезке стирается.

1. **Код программы**

Основная программа:

**uses** GraphABC;

**Uses** Build;

**var** i: integer;

**var** x1, y1, x2, y2, x3, y3, l, t: integer;

**procedure** text;

**begin**

textout(10, 100, 'Left: влево');

textout(10, 120, 'Right: вправо');

textout(10, 140, 'Up: вверх');

textout(10, 160, 'Down: вниз');

textout(10, 180, 'W: увеличение');

textout(10, 200, 'S: уменьшение');

textout(10, 220, 'E: + 1 итерация(до 5)');

textout(10, 240, 'Q: - 1 итерация(до 0)');

**end**;

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

**begin**

i := 0;

**case** key **of**

VK\_Up: **begin** y1 := y1 - 15; y2 := y2 - 15; y3 := y3 - 15 **end**;

VK\_Down: **begin** y1 := y1 + 15; y2 := y2 + 15; y3 := y3 + 15 **end**;

VK\_Left: **begin** x1 := x1 - 15; x2 := x2 - 15; x3 := x3 - 15 **end**;

VK\_Right: **begin** x1 := x1 + 15; x2 := x2 + 15; x3 := x3 + 15 **end**;

VK\_E: **begin** t += 1; **if** t=6 **then** t:=5; **end**;

VK\_Q: **begin** t -= 1; **if** t=-1 **then** t:=0; **end**;

VK\_W:

**begin**

x1 := Round(x1 \* 1.1);

x2 := Round(x2 \* 1.1);

x3 := Round(x3 \* 1.1);

y1 := Round(y1 \* 1.1);

y2 := Round(y2 \* 1.1);

y3 := Round(y3 \* 1.1);

l:=round(l\*1.1)

**end**;

VK\_S:

**begin**

x1 := Round(x1 / 1.1);

x2 := Round(x2 / 1.1);

x3 := Round(x3 / 1.1);

y1 := Round(y1 / 1.1);

y2 := Round(y2 / 1.1);

y3 := Round(y3 / 1.1);

l:=round(l/1.1)

**end**;

**end**;

Window.Clear;

text;

Draw(x1, y1, l, pi / 3, t);

Draw(x2, y2, l, pi, t);

Draw(x3, y3, l, -pi / 3, t);

redraw;

**end**;

**begin**

SetWindowSize(850, 500);

SetWindowCaption('Фрактал: Снежинка Коха');

Text;

LockDrawing;

x1:=410; x2:= 810; x3:=610;

y1:=354; y2:=354; y3:=8;

t:=0;

l:=400;

Draw(x1, y1, l, pi / 3, t);

Draw(x2, y2, l, pi, t);

Draw(x3, y3, l, -pi / 3, t);

redraw;

onKeyDown += KeyDown;

**end**.

Модуль:

**Unit** Build;

**Interface**

**Uses** GraphABC;

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

**implementation**

**var** i: integer;

**var** x1, y1, x2, y2, x3, y3, dx1, dy1, dx2, dy2, dx3, dy3, l, t: integer;

**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**;

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

**begin**

i := 400;

**case** key **of**

VK\_Up: **begin** y1 := y1 - 15; y2 := y2 - 15; y3 := y3 - 15 **end**;

VK\_Down: **begin** y1 := y1 + 15; y2 := y2 + 15; y3 := y3 + 15 **end**;

VK\_Left: **begin** x1 := x1 - 15; x2 := x2 - 15; x3 := x3 - 15 **end**;

VK\_Right: **begin** x1 := x1 + 15; x2 := x2 + 15; x3 := x3 + 15 **end**;

VK\_E: **begin** t += 1; **if** t=6 **then** t:=5; **end**;

VK\_Q: **begin** t -= 1; **if** t=-1 **then** t:=0; **end**;

VK\_W:

**begin**

i := i + 40;

x1 := x1 + i;

x2 := x2 + i;

x3 := x3 + i;

y1 := y1 + i;

y2 := y2 +i;

y3 := y3 + i;

**end**;

VK\_S:

**begin**

i := i - 40;

x1 := x1 - i;

x2 := x2 - i;

x3 := x3 - i;

y1 := y1 - i;

y2 := y2 - i;

y3 := y3 - i;

**end**;

**end**;

**end**;

**Begin**

SetWindowSize(850, 500);

SetWindowCaption('Фрактал: Снежинка Коха');

LockDrawing;

KeyDown(0);

OnkeyDown += KeyDown;

**end**.

1. **Результат выполнения программы**

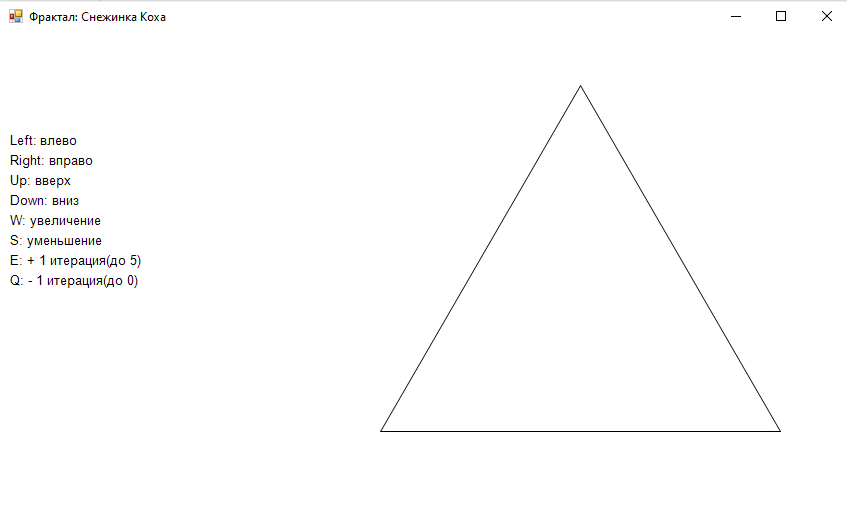


Рисунок 1 – Начальный фрактал

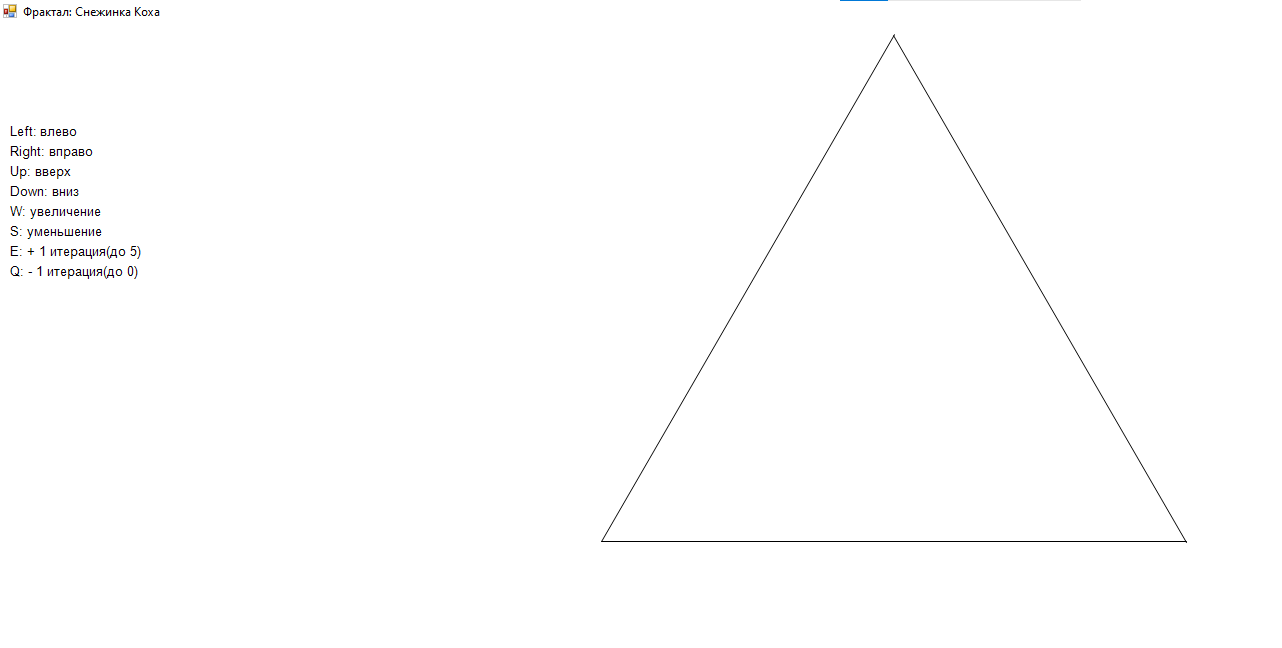


Рисунок 2 – изменение масштаба(увеличение)

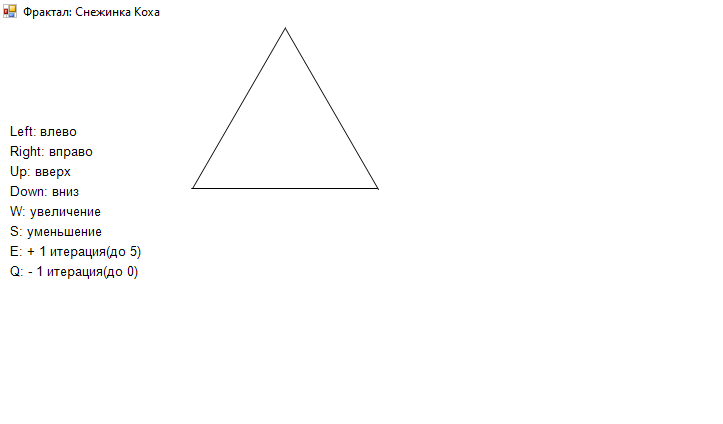


Рисунок 3 – изменение масштаба(уменьшение)

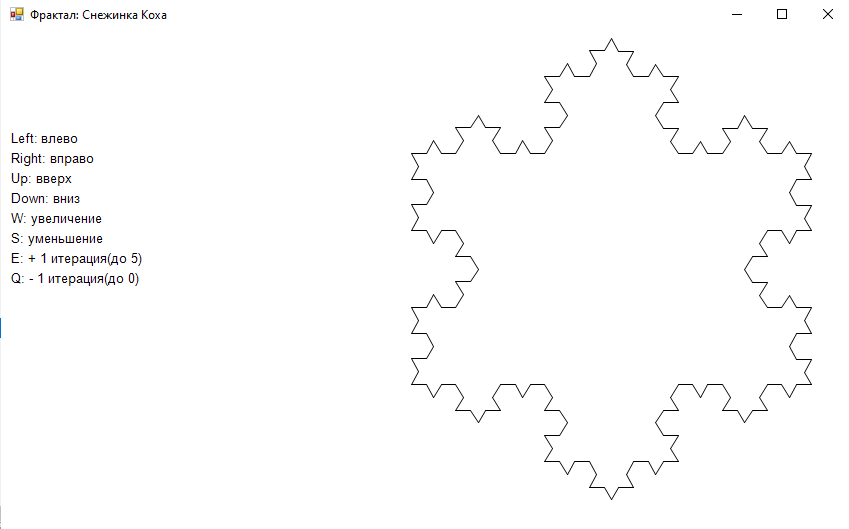


Рисунок 4 – Увеличение количества итераций

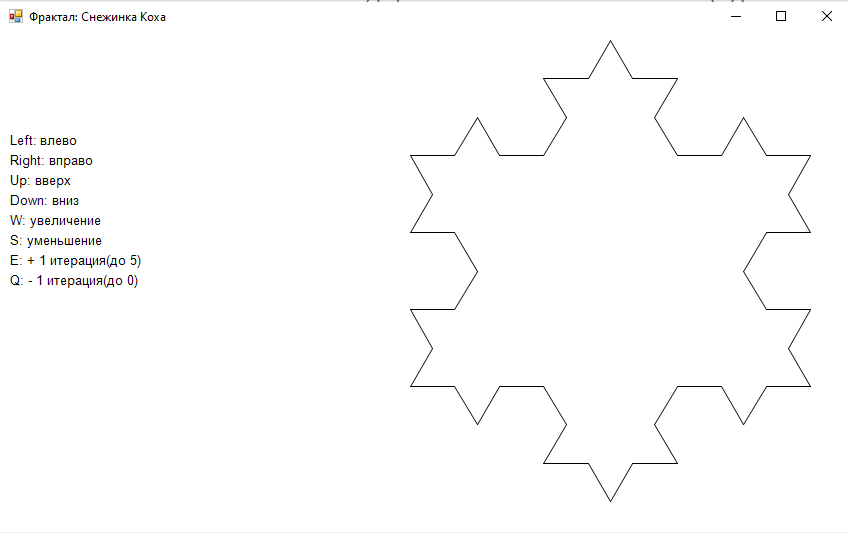


Рисунок 5 – Уменьшение количества итераций

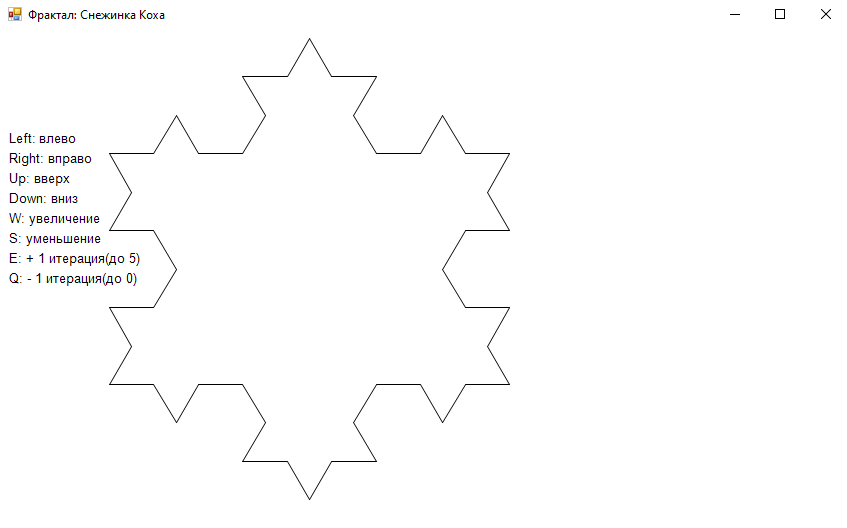


Рисунок 6 – Перемещение фрактала по области

1. **Вывод**

В ходе данной домашней контрольной работы были получены знания о визуализации фрактала, его перемещении по области и масштабирования. Так же были получены знания в создании модулей в языке программирования Pascal.

Еще были получены умения в реализации кнопок как интерактивная часть.