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**

**«\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_»**

**ПО МДК 05.02 РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ**

Выполнил: студент учебной группы

ИСПк-205-52-00

Подоплелов Дмитрий Андреевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

В отчете должны отображаться:

1. Цель работы
2. Скриншот задания (с вариантом)
3. Описание алгоритма
4. Код программы
5. Результат выполнения программы
6. Вывод

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

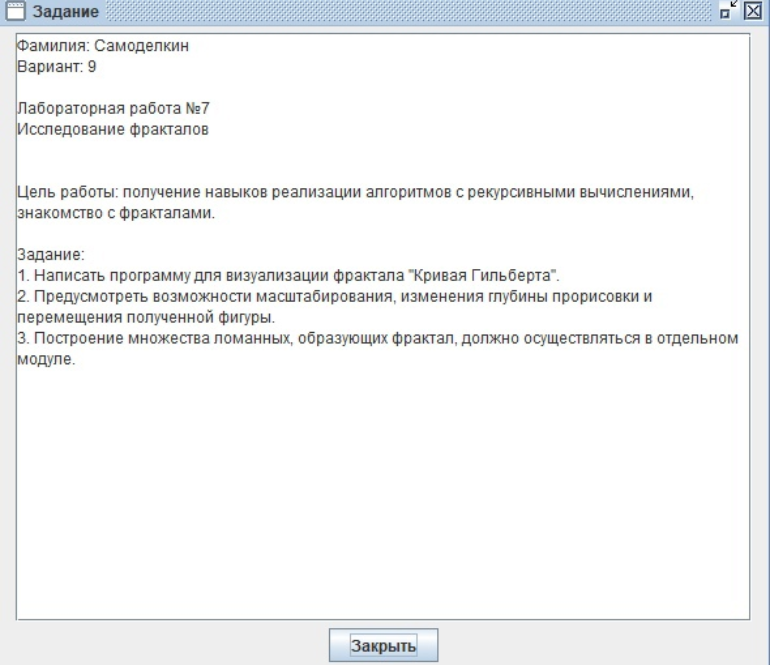


Рисунок 1. Задание

Кривая Гильберта (известная также как заполняющая пространство кривая Гильберта) — это непрерывная [фрактальная](https://ru.wikipedia.org/wiki/%D0%A4%D1%80%D0%B0%D0%BA%D1%82%D0%B0%D0%BB) [заполняющая пространство кривая](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%BF%D0%BE%D0%BB%D0%BD%D1%8F%D1%8E%D1%89%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D1%81%D1%82%D1%80%D0%B0%D0%BD%D1%81%D1%82%D0%B2%D0%BE_%D0%BA%D1%80%D0%B8%D0%B2%D0%B0%D1%8F), впервые описанная немецким математиком [Давидом Гильбертом](https://ru.wikipedia.org/wiki/%D0%93%D0%B8%D0%BB%D1%8C%D0%B1%D0%B5%D1%80%D1%82,_%D0%94%D0%B0%D0%B2%D0%B8%D0%B4).

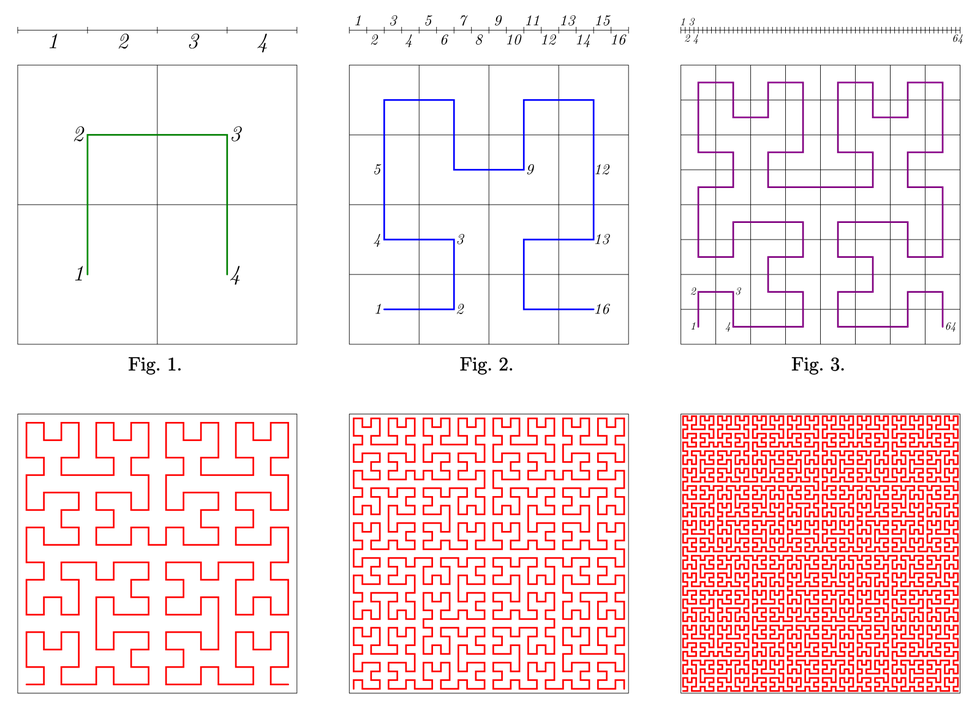


Рисунок 2. Кривая Гилберта алгоритм решения

**Uses** GraphWPF;

**var** (u,p, wx,wy) := (32,4, 64,64);

**Procedure** a(i: Integer); **forward**;

**Procedure** b(i: Integer); **forward**;

**Procedure** c(i: Integer); **forward**;

**Procedure** d(i: Integer); **forward**;

**Procedure** a(i: Integer) :=

**If** i > 0 **Then begin**

d(i - 1); LineBy(+u, 0);

a(i - 1); LineBy(0, u);

a(i - 1); LineBy(-u, 0);

c(i - 1)

**end**;

**Procedure** b(i: integer) :=

**If** i > 0 **Then begin**

c(i - 1); LineBy(-u, 0);

b(i - 1); LineBy(0, -u);

b(i - 1); LineBy(u, 0);

d(i - 1)

**end**;

**Procedure** c(i: integer) :=

**If** i > 0 **Then begin**

b(i - 1); LineBy(0, -u);

c(i - 1); LineBy(-u, 0);

c(i - 1); LineBy(0, u);

a(i - 1)

**end**;

**Procedure** d(i: integer) :=

**If** i > 0 **Then begin**

a(i - 1); LineBy(0, u);

d(i - 1); LineBy(u, 0);

d(i - 1); LineBy(0, -u);

b(i - 1)

**end**;

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

**begin**

**case** K **of**

Key.Down: wy -= 10;

Key.Up: wy += 10;

Key.Left: wx += 10;

Key.Right: wx -= 10;

Key.Escape: halt(1); //выход

Key.A: **if** p\*u < 1000 **then** u := p\*u>50 ? (u \* 1.1).trunc : u+5;

Key.Z: **if** (p\*u > 50) **and** (u > 9) **then** u := (u / 1.1).trunc;

Key.S: **if** (p\*u > 50) **and** (p < 5) **then** (u,p) :=(u **div** 2, p+1);

Key.X: **if** p > 1 **then** (u,p) :=(u\*2, p-1);

**end**;

MoveTo(wx,wy);

Redraw(() ->**begin** Window.Clear; a(p); **end**);

**end**;

**begin**

Window.SetSize(650,650);

MoveTo(wx, wy);

Redraw(() ->a(p));

OnkeyDown += KeyDown;

**end**.

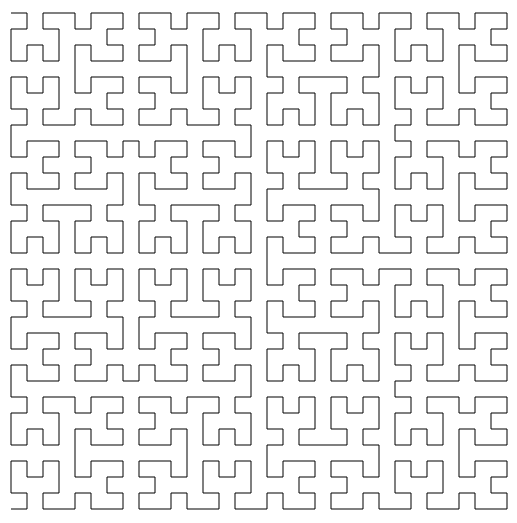


Рисунок 3. Результат выполнения программы.

Вывод: В результате написания программы был изучен фрактал “Кривая Гильберта”. Был написан код и в дальнейшем визуализирован. Была добавлена возможность масштабирования, изменение глубины прорисовки и перемещения полученной фигуры. В ходе выполнения программы проблем не возникло.