ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
"ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ"

Факультет ИСП

Кафедра программной инженерии им. Л.П. Фельдмана

# ОТЧЁТ

к лабораторной работе по дисциплине  
«Профессиональная практика программной инженерии»  
на тему: «Создание самодокументирующегося кода»

Выполнил:

ст. гр. ПИ-19б  
Муращенко А. Р.

Проверил:

ассистент каф.

Дмитрюк Т.Г.

Донецк 2023

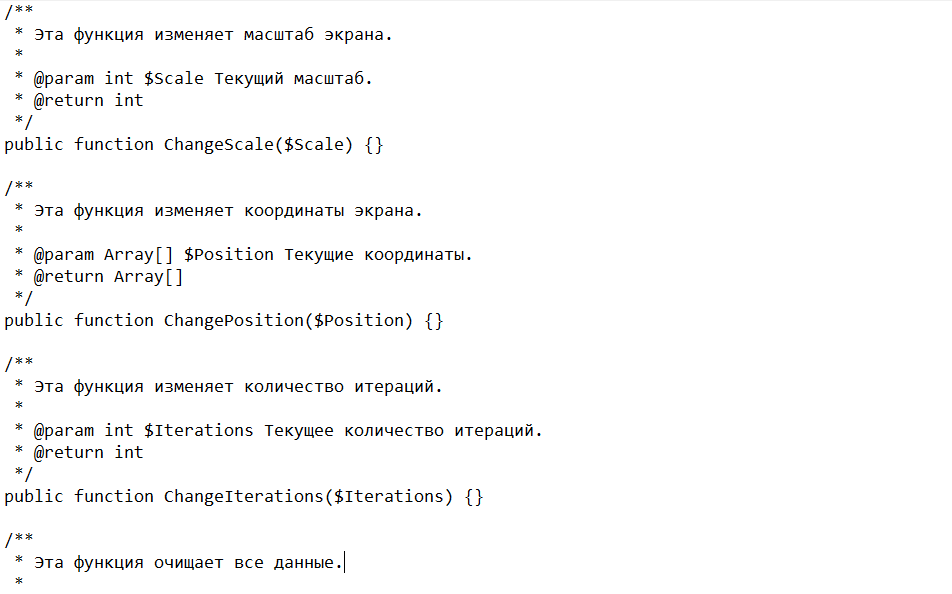
Цель работы: научиться добавлять в программный код специальным образом оформление докблок-комментарии, для последующей автоматической генерации API reference, а также познакомиться с форматом оформления документации DocBook.

Задание к лабораторной работе

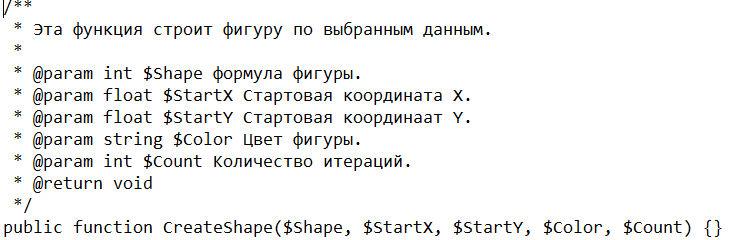
Внедрить в ранее разработанный программный код докблоки. Разработать DocBook и проверить корректность созданного DocBook файла с помощью средств редактора или одного из онлайн валидаторов..

Ход работы

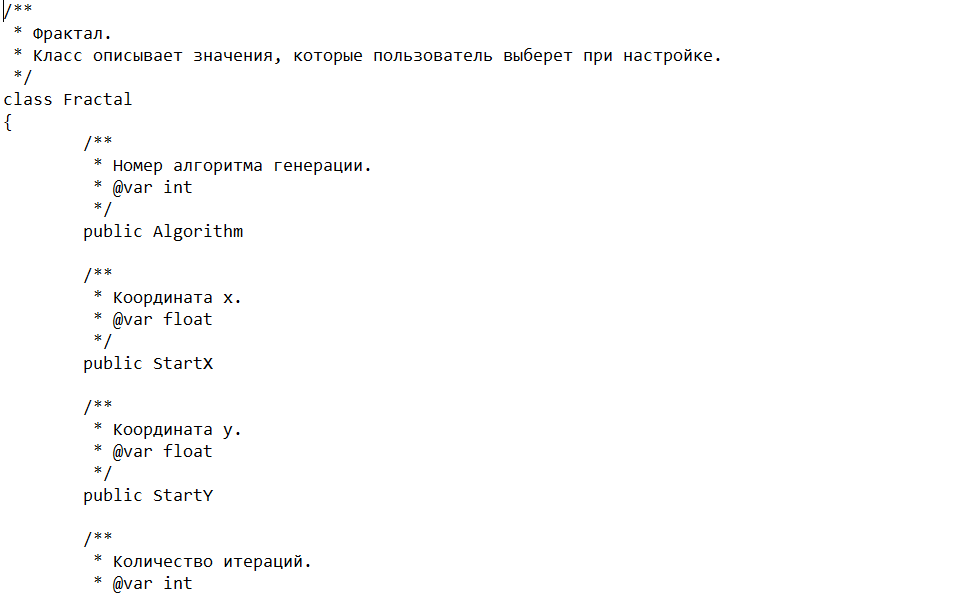
Создаём докблоки в файлах программы.

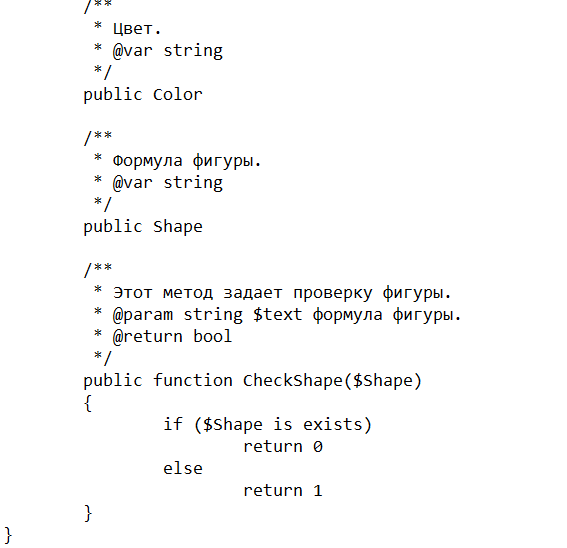
Содержимое файла Control.txt

Содержимое файла Create.txt

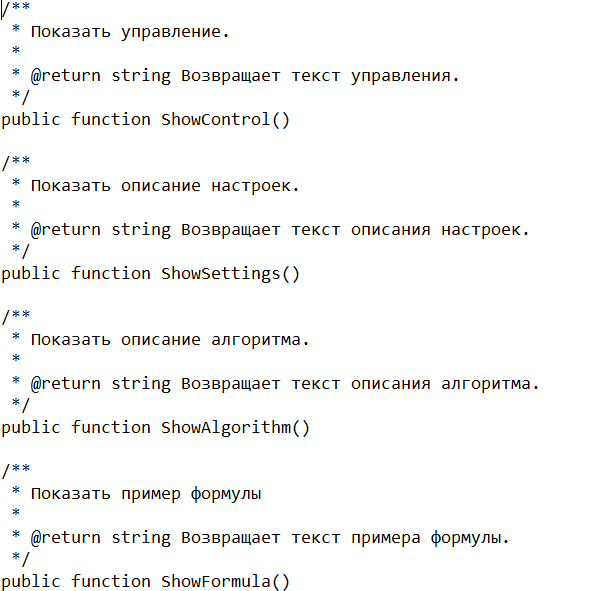


Содержимое Generation.txt

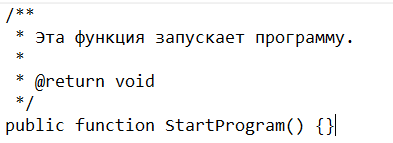




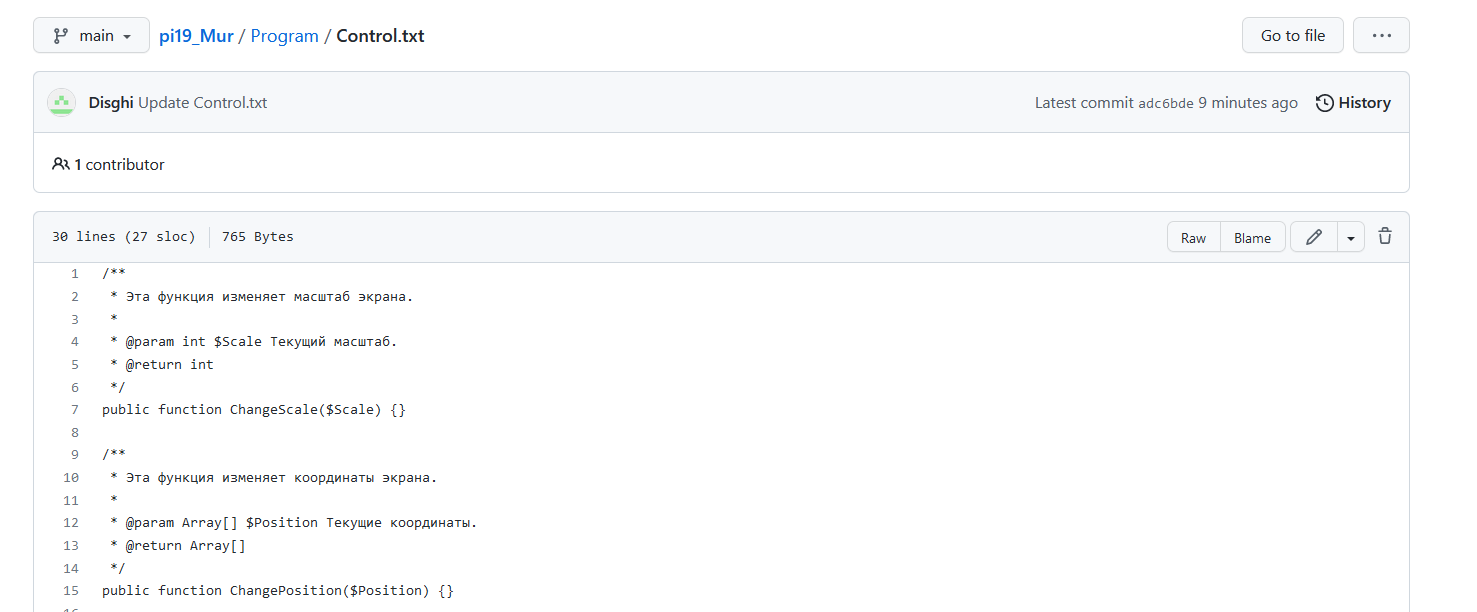
Содержимое файла help.txt



Содержимое файла Main.txt



Добавляем изменения в репозиторий:



Создаём краткое описание программы:

Генератор фракталов, с реализацией базовых алгоритмов построения фрактальных изображений (алгоритмические, геометрические и фракталы на основе метода IFS).

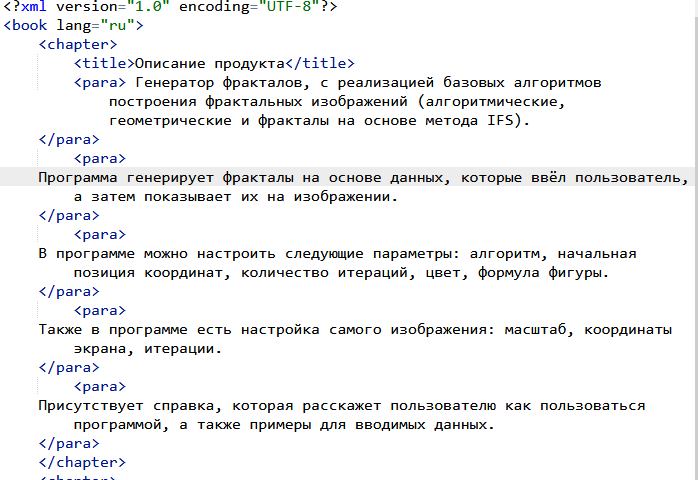
Программа генерирует фракталы на основе данных, которые ввёл пользователь, а затем показывает их на изображении.

В программе можно настроить следующие параметры: алгоритм, начальная позиция координат, количество итераций, цвет, формула фигуры.

Также в программе есть настройка самого изображения: масштаб, координаты экрана, итерации.

Присутствует справка, которая расскажет пользователю, как пользоваться программой, а также примеры для вводимых данных.

Добавляем описание в DocBook:



Сценарии использования:

Сценарий 1

1. Пользователь запускает программу с помощью функции StartProgram().

2. Пользователь смотрит примеры использования в справке. ShowControl(); ShowSettings(); ShowAlgorithm(); ShowFormula().

3. Пользователь настривает параметры.

4. Программа генерирует фракталы. CreateShape($Shape, $StartX, $StartY, $Color, $Count).

5. Пользователь закрывает программу.

Сценарий 2

1. Пользователь запускает программу с помощью функции StartProgram().

2. Пользователь настривает параметры.

3. Программа генерирует фракталы. CreateShape($Shape, $StartX, $StartY, $Color, $Count).

4. Пользователь изменяет масштаб экрана. ChangeScale($Scale).

5. Пользователь изменяет координату экрана. ChangePosition($Position).

6. Пользователь закрывает программу.

Сценарий 3

1. Пользователь запускает программу с помощью функции StartProgram().

2. Пользователь настривает параметры.

3. Программа генерирует фракталы. CreateShape($Shape, $StartX, $StartY, $Color, $Count).

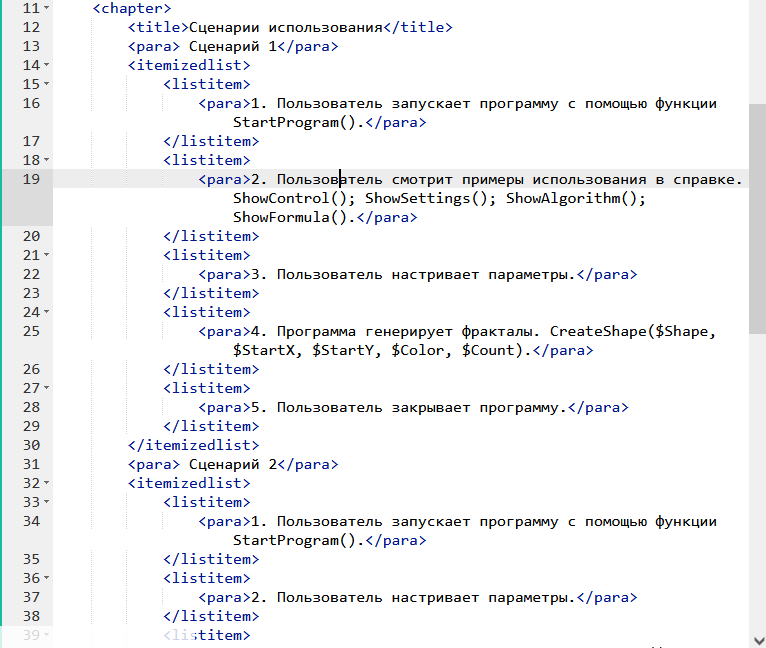
4. Пользователь очищает все данные. ClearAll().

5. Пользователь заново вводит все параметры.

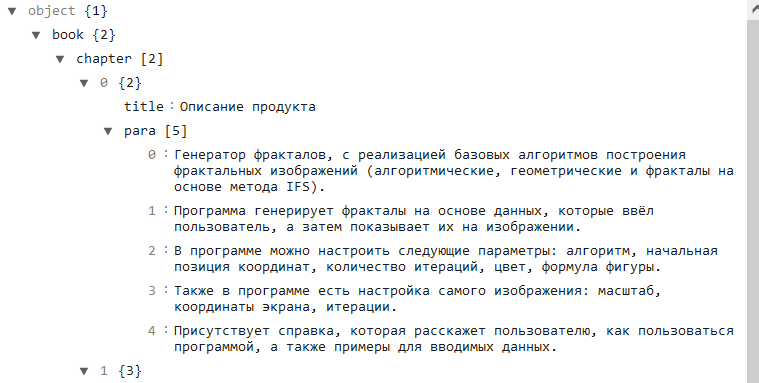
6. Программа генерирует заново фракталы.

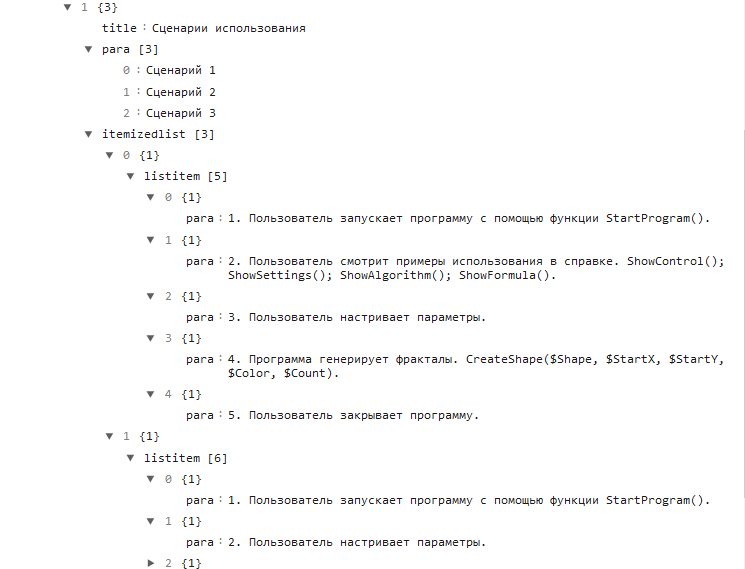
7. Пользователь закрывает программу.

Добавляем эти сценарии в DocBook:



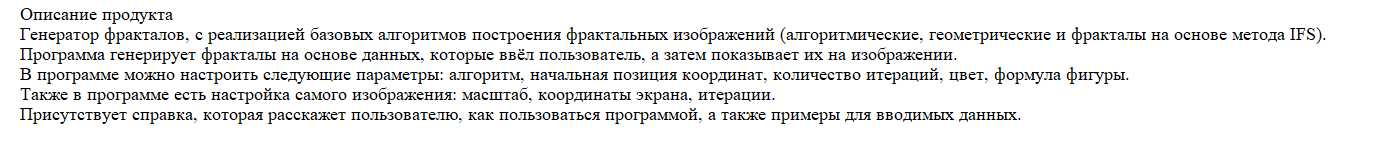
Итоговая структура DocBook:

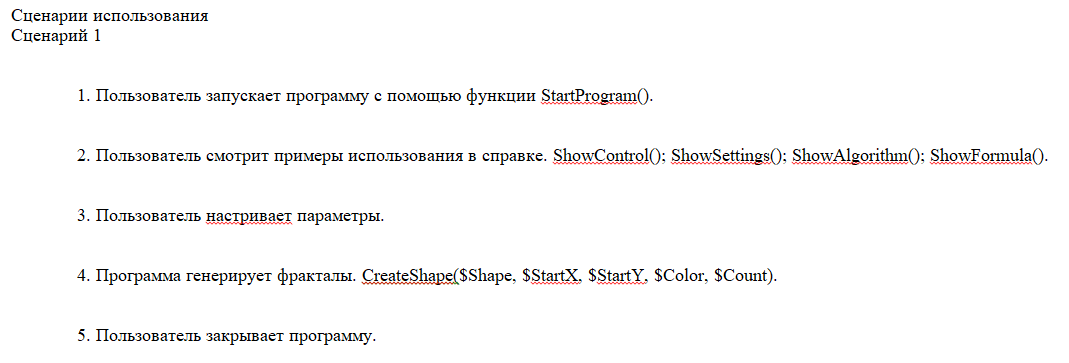


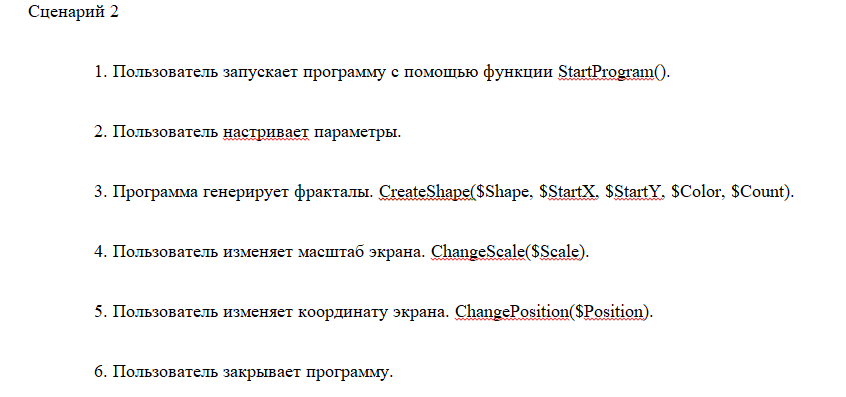


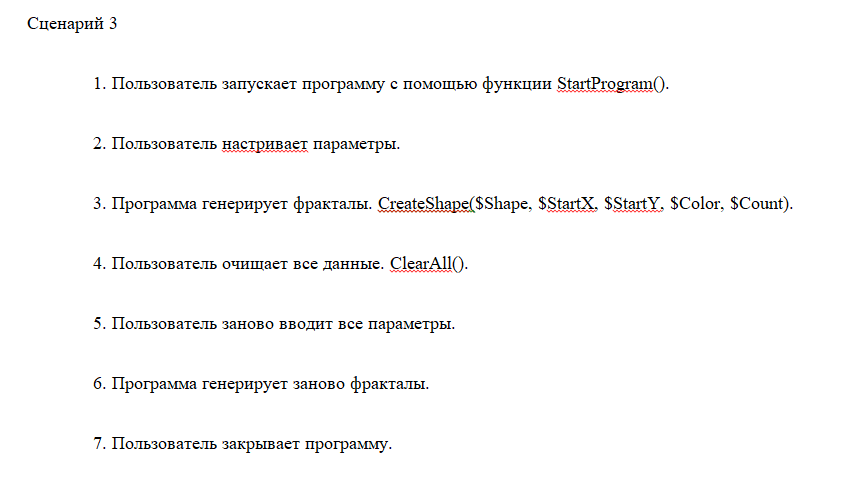
Онлайн валидатор jsonformatter.org ошибок не нашёл.

Открытие xml файла через Word:









Вывод: На лабораторной работе научились вставлять докблоки в код программы, а также составлять документ на DocBook. Я сделал докблоки к 5 файлам программы, а также составил краткое описание программы и 3 сценария использования.