ФЕДЕРАЛЬНОЕ АГЕНСТВО СВЯЗИ

Ордена Трудового Красного Знамени

федеральное государственное бюджетное

образовательное учреждение высшего образования

«Московский Технический Университет Связи и Информатики» (МТУСИ)

Кафедра МКиИТ

Лабораторная работа №6

«MultiCoreFractalGenerator»

Выполнила:

Студентка 2 курса

Группы БСТ1802

Матюшина Алина

Проверил:

Городничев М. Г.

Москва 2020

Оглавление

[Цель лабораторной работы: 3](file:///C:\Users\misgood\Documents\GitHub\bulanov\Лабораторная%20работа%20№3\Лабораторная%20работа%20№3.docx#_Toc507946798)

[Задачи: 3](file:///C:\Users\misgood\Documents\GitHub\bulanov\Лабораторная%20работа%20№3\Лабораторная%20работа%20№3.docx#_Toc507946799)

[Анализ предметной области и выбор инструментария: 3](file:///C:\Users\misgood\Documents\GitHub\bulanov\Лабораторная%20работа%20№3\Лабораторная%20работа%20№3.docx#_Toc507946800)

[Объяснение функций: 3](file:///C:\Users\misgood\Documents\GitHub\bulanov\Лабораторная%20работа%20№3\Лабораторная%20работа%20№3.docx#_Toc507946801)

[Вывод: 3](file:///C:\Users\misgood\Documents\GitHub\bulanov\Лабораторная%20работа%20№3\Лабораторная%20работа%20№3.docx#_Toc507946802)

# Цель лабораторной работы:

Закончить в этой работе с генератором фракталов с еще одной функцией – возможностью нарисовать фрактал с многопоточными фоновыми процессами.

# Задачи:

Внести изменения, используя встроенную поддержку Swing для фоновых потоков.

# Анализ предметной области и выбор инструментария:

В ходе работы были использованы пакет разработчика приложений Java JDK, командная строка и NotePad++.

# Объяснение функций:

• FractalExplorer.java - позволяет исследовать различные части фрактала, создавая и показывая GUI Swing и обрабатывая события, вызванные различным взаимодействием с пользователем.

• FractalGenerator.java – предоставленный исходный файл, из которого происходят наши фрактальные генераторы.

• JImageDisplay.java – графический виджет, позволяющий отображать фракталы.

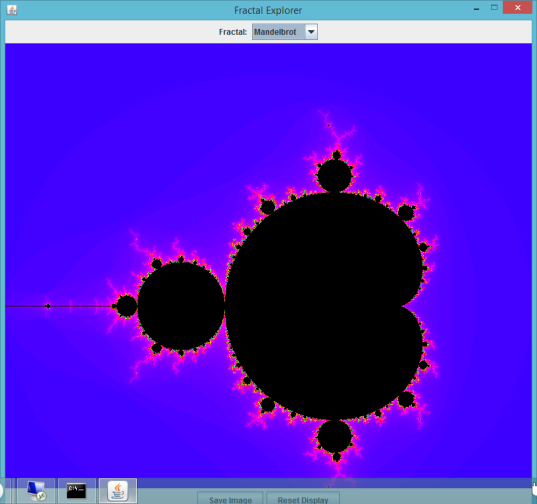
• Mandelbrot.java – производит итерации для построения фрактала

•Tricorn.java – практически идентичная имплементация фракталу Мандельброта, за исключением равенства zn = zn-1 2 + c. Единственное отличие в том, что берется сложное сопряжение zn-1. Каждая итерация начинается с изначальной области определения фрактала tricorn и является от (-2, -2) до (2,2).

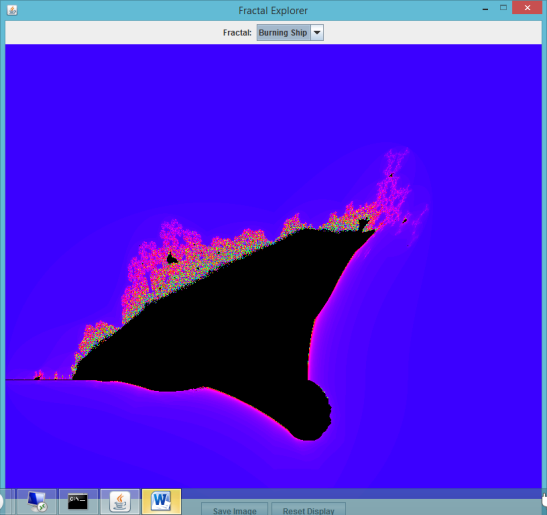
•BurningShip – второй фрактал, который имплементирован, это «горящий корабль», который так назван, потому что очень похож на горящий корабль. Вот детали: Равенство zn = (|Re(zn-1)| + i |Im(zn-1)|)2 + c. Другими словами, берется абсолютное значение каждого компонента zn-1. Каждая итерация начинается с изначальной области определения данного фрактала, т.е. от (-2, -2.5) до (2,1.5).

# Результат работы программы:

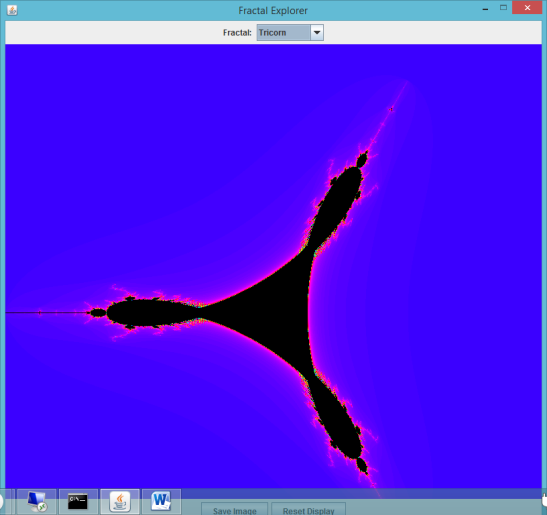
Фрактал Mandelbrot:



Фрактал Burning Ship:



Фрактал Tricorn:



# Вывод:

В ходе выполнения работы программа была изменена так, чтобы она использовала одно или несколько фоновых ядер для вычисления фрактала. В итоге получилась красивая программа отрисовки фракталов, которая может рисовать фракталы с несколькими потоками и это не позволит пользователям ничего делать, пока процесс рендеринга происходит в фоновом режиме.