

Разработка программного обеспечения для моделирования изображения трехмерного фрактала «оболочка Мандельброта»

Студент: Варченко М. А., ИУ7-51Б

Руководитель: Волкова Л. Л.

2024 г.

Цель и задачи

Цель – разработка программного обеспечения для моделирования изображения трехмерного фрактала «оболочка Мандельброта».

Задачи:

- проанализировать существующие методы и алгоритмы для моделирования изображений аналитических поверхностей;
- выбрать и описать алгоритмы реализации, необходимые для визуализации фрактала;
- реализовать выбранные алгоритмы;
- разработать интерфейс для управления параметрами сцены, изображаемого объекта и положением камеры;
- провести исследование производительности программы при разных параметрах сцены.

Визуализируемая сцена

Сцена состоит из следующего набора объектов:

- 1) фрактал (изображаемый объект);
- 2) точечный источник света;
- 3) камера;

Требования:

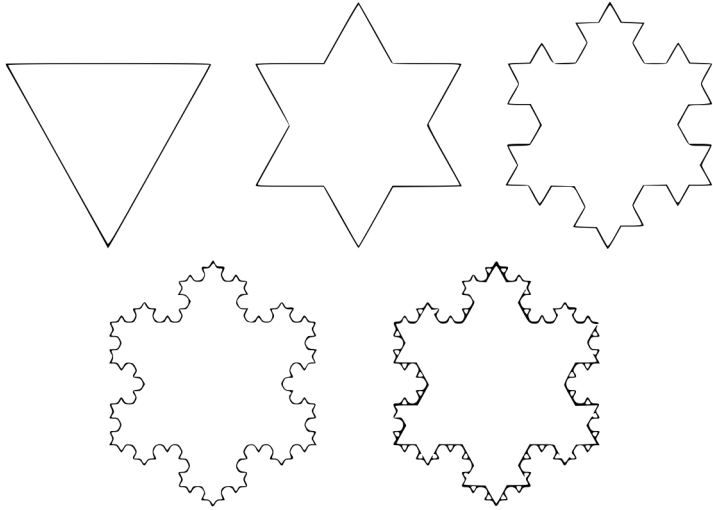
- программа должна предоставлять возможность изменять параметры модели (степень фрактала, количество итераций для построения, радиус отсечения);
- должна быть возможность изменения параметров сцены (цвет фона, цвет объекта) и источника света (интенсивность, положение);
- пользователь должен иметь возможность управления камерой.

Определение и классификация фракталов

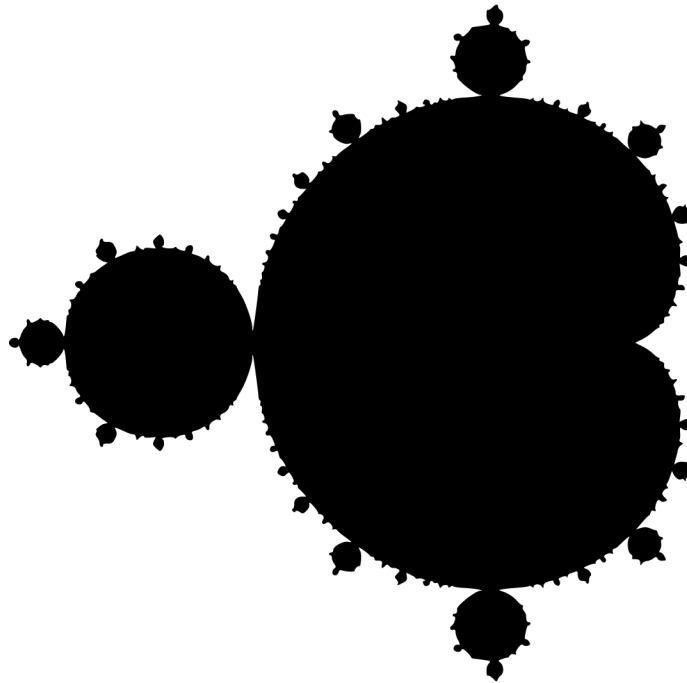
Фрактал - это структура, состоящая из частей, которые в каком-то смысле подобны целому.

Фракталы делят на:

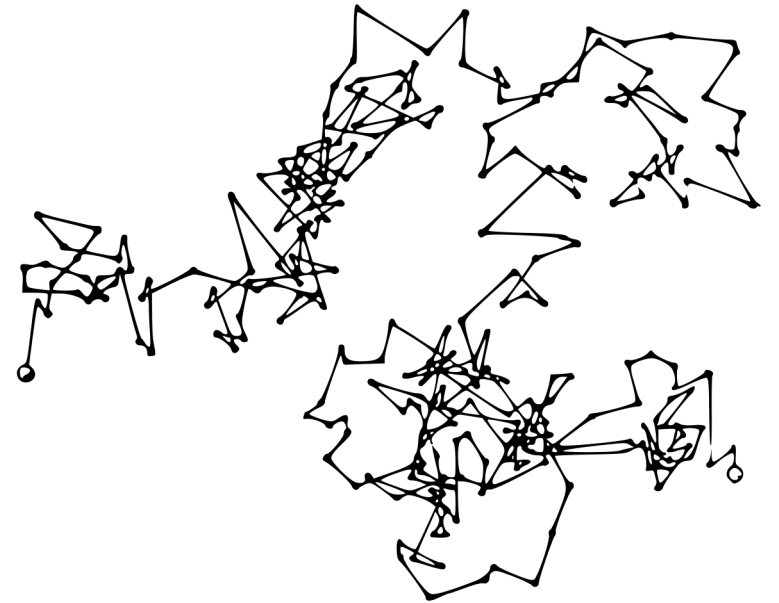
геометрические



алгебраические



стохастические



Оболочка Мандельброта

Оболочка Мандельброта – трехмерный фрактал, аналог множества Мандельброта в трехмерном пространстве с использованием гиперкомплексной алгебры.

Для построения используется итерация $z \mapsto z^n + c$.

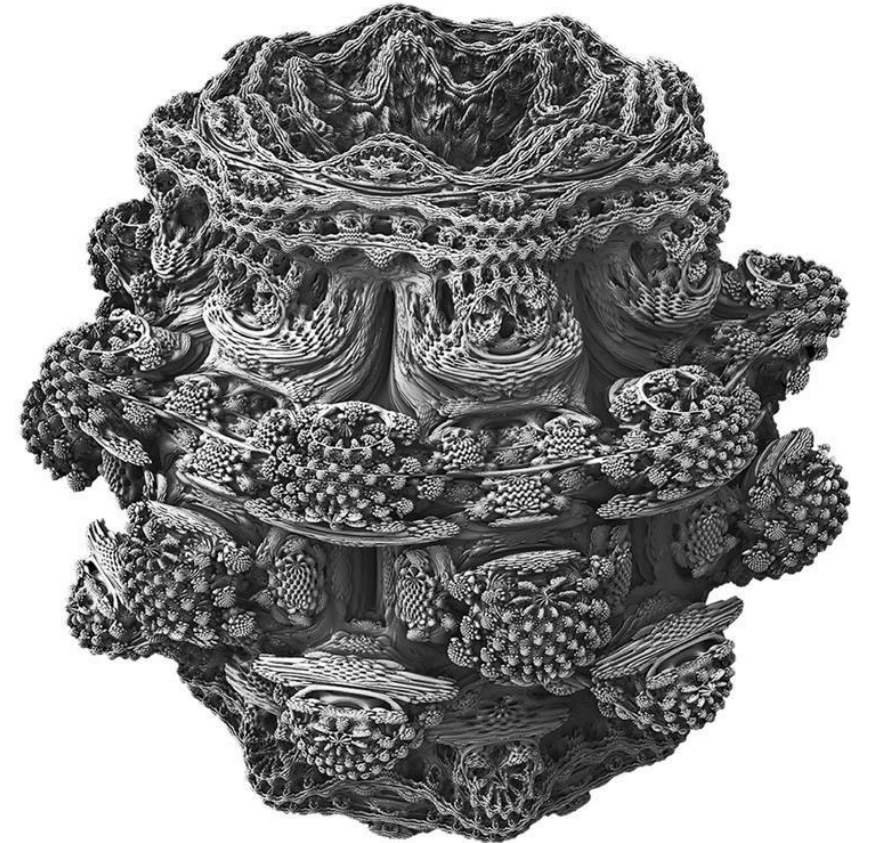
Формула для n степени трехмерного гиперкомплексного числа:

$$\langle x, y, z \rangle^n = r^n \langle \cos(n\theta)\cos(n\phi), \sin(n\theta)\cos(n\phi), \sin(n\phi) \rangle,$$

где $r = \sqrt{x^2 + y^2 + z^2};$

$$\theta = \arctan(y/x);$$

$$\phi = \arctan(z/\sqrt{x^2 + y^2}) = \arcsin(z/r);$$



Анализ и выбор метода рендера модели и метода закраски

Метод	Растеризация	Ray Tracing	Ray Marching
Качество	3	1	2
Быстродействие	1	3	2
Сложные объекты	-	-	+

Метод	Простой метод	Метод Гуро	Метод Фонга
Качество	3	2	1
Быстродействие	1	2	3

Поля расстояний со знаком

Поля расстояний со знаком (SDF) используются для нахождения кратчайшего расстояния от точки в пространстве до поверхности тела.

Знак результата определяет положение точки относительно данного объекта:

- вне объекта, если $SDF(p) > \varepsilon$;
- на границе объекта, если $|SDF(p)| \leq \varepsilon$;
- внутри объекта, если $SDF(p) < -\varepsilon$;

где $\varepsilon \rightarrow 0$ - допустимая погрешность.

Поле расстояний со знаком для сферы:

$$SDF_{sphere}(x, y, z) = \sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2} - r,$$

где $C(x_0, y_0, z_0)$ — координаты центра сферы;

r — радиус сферы.

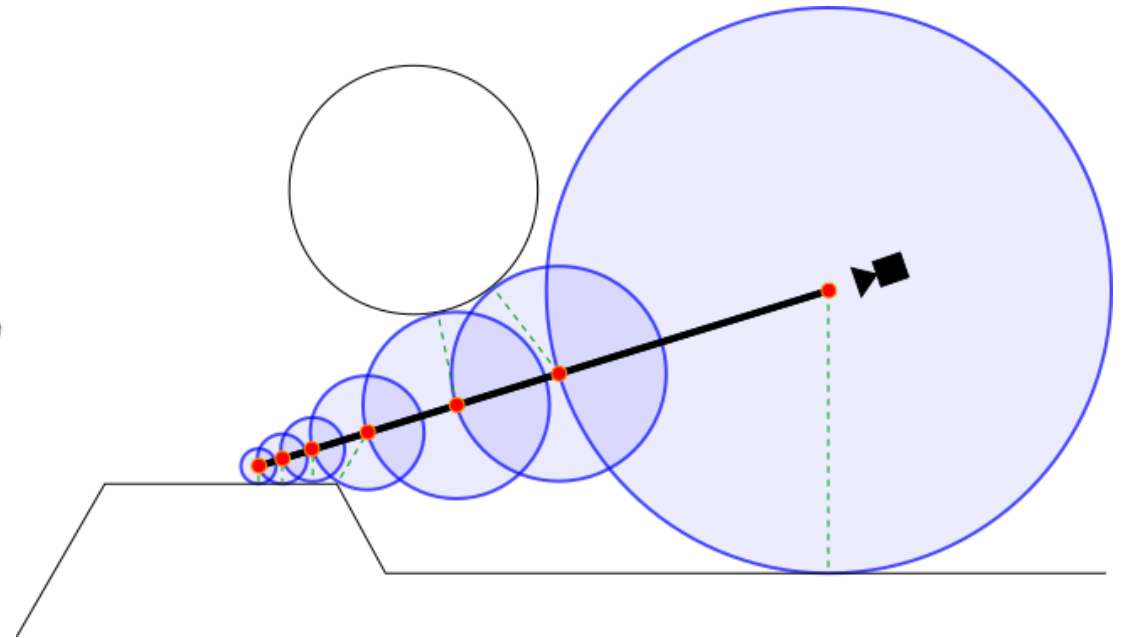


Схема алгоритма SDF для оболочки Мандельброта

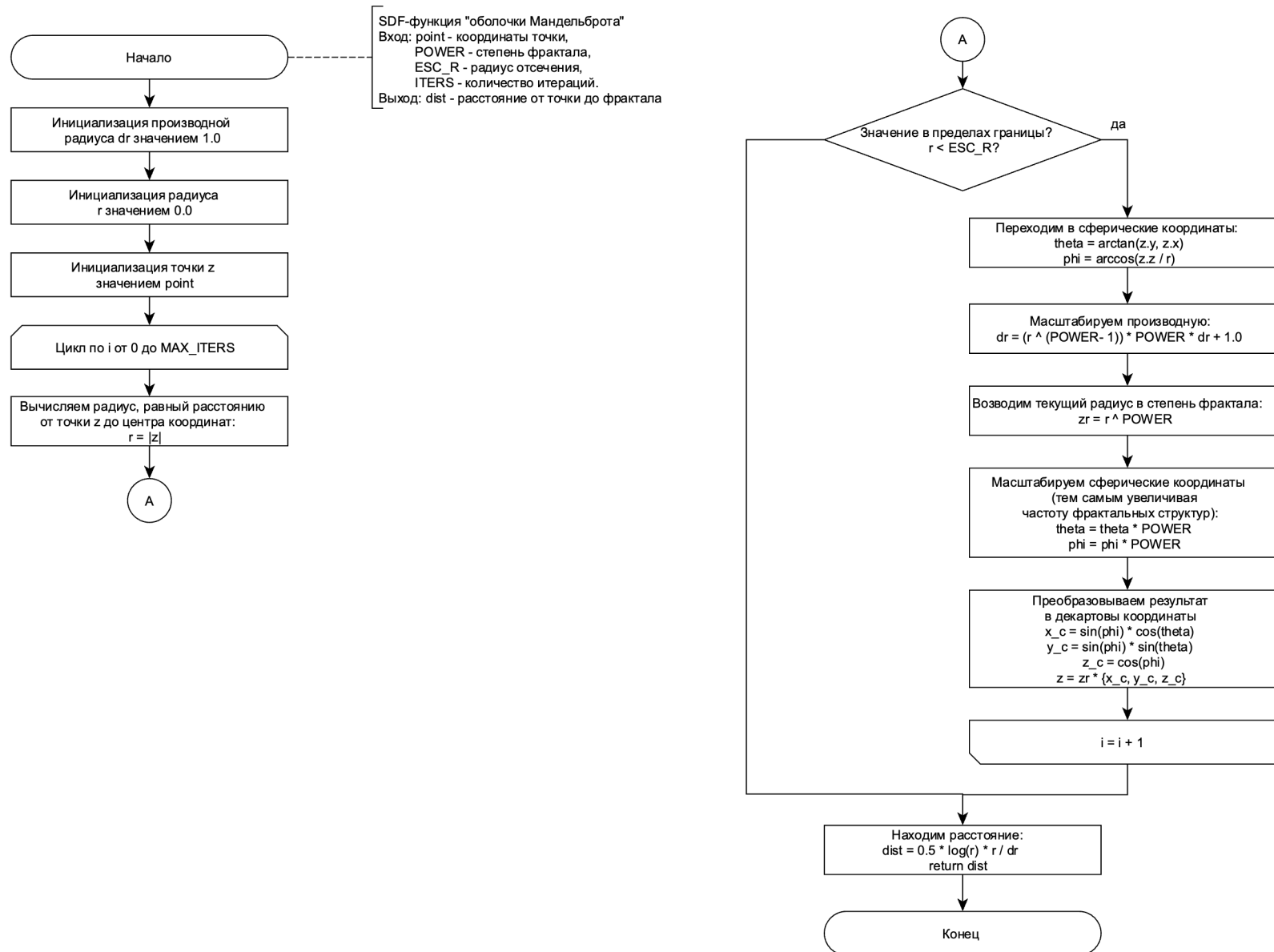
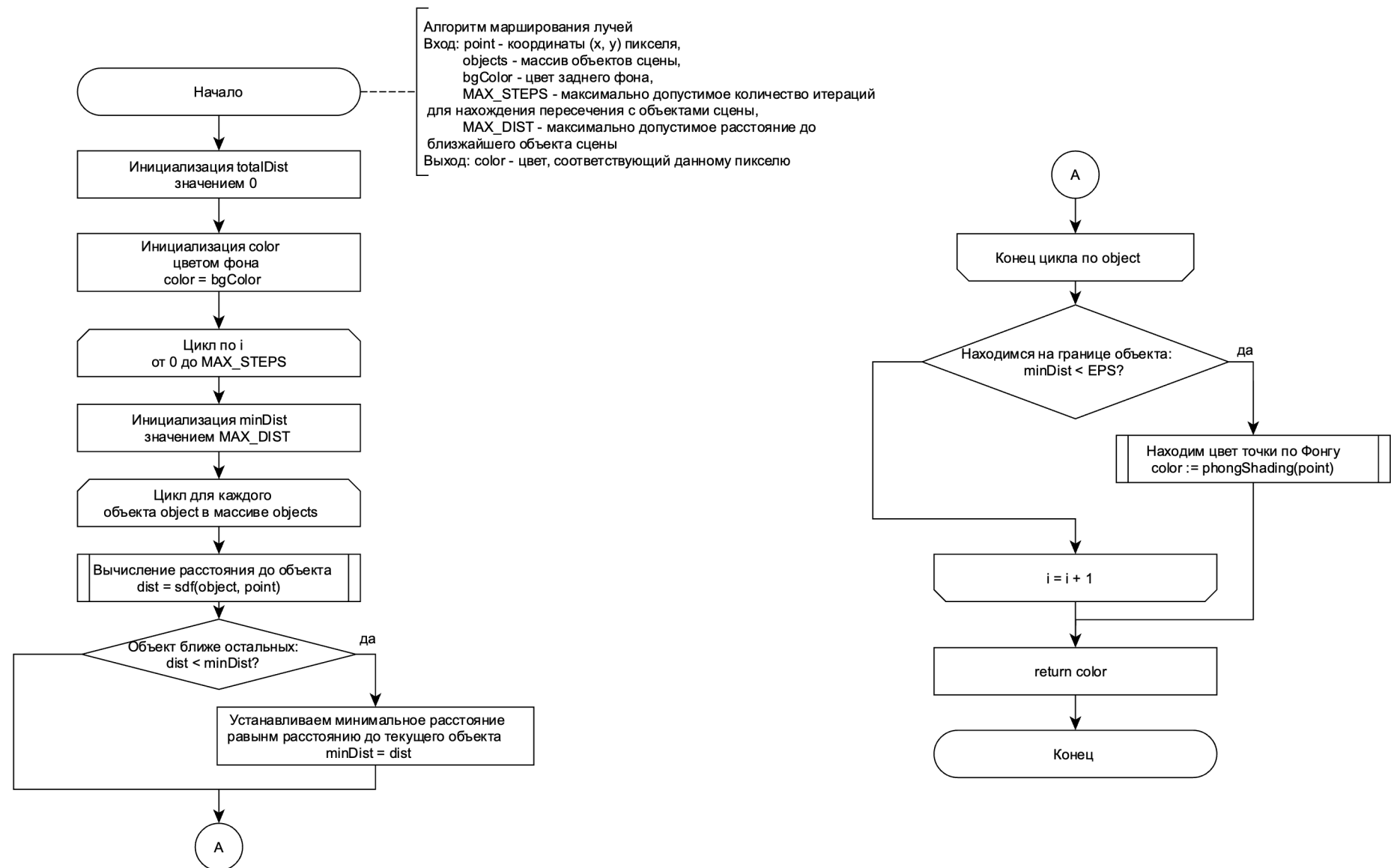


Схема алгоритма марширования лучей



Выбор средств реализации

В качестве основного языка программирования выбран C++:

- поддержка объектно-ориентированной парадигмы;
- высокая производительность;
- наличие строгой типизации.

Для реализации графического интерфейса выбран фреймворк Qt:

- кроссплатформенность;
- встроенные инструменты для работы с трехмерными объектами.

Для работы с графическим процессором использована библиотека OpenGL:



- кроссплатформенность;
- наличие необходимого для решения поставленной задачи функционала.

Интерфейс программы

- Кнопки «Цвет объекта/фона» открывают диалоговое окно с выбором нового цвета;
- Меню «выбор объекта» позволяет выбрать моделируемый объект из выпадающего списка;
- Меню «выбор модели рендера» позволяет выбрать алгоритм рендера из выпадающего списка;
- Меню «выбор модели закраски» позволяет выбрать алгоритм закраски из выпадающего списка;
- Поле «интенсивность» отвечает за изменение интенсивности источника света;
- Кнопка «совместить с позицией камеры» позволяет совместить источник света с камерой;
- Поля «x», «y», «z» позволяют вручную задать положение источника света в пространстве;

В зависимости от вида моделируемого объекта, в нижней части интерфейса отображаются параметры, характерные для данного вида объектов.

The image shows a settings dialog box for a 3D rendering application. It is organized into several sections with titles in Russian. The first section, 'Настройки цвета' (Color Settings), has two columns: 'Цвет объекта' (Object Color) with a teal square and 'Цвет фона' (Background Color) with a gray square. The next three sections are dropdown menus: 'Выбор объекта' (Object Selection) set to 'Оболочка Мандельброта' (Mandelbrot Shell), 'Выбор модели рендера' (Renderer Selection) set to 'Марширование лучей (OpenGL)' (Ray Tracing (OpenGL)), and 'Выбор модели затенения' (Shading Model Selection) set to 'Затенение по Фонгу' (Fog Shading). The 'Выбор режима камеры' (Camera Mode Selection) dropdown is set to 'Вращение вокруг объекта' (Rotate around object). The 'Настройки источника освещения' (Light Source Settings) section includes an 'Интенсивность' (Intensity) field set to 1.0, a checked checkbox 'Совместить с позицией камеры' (Align with camera position), and three coordinate fields (x, y, z) all set to 4.0, with an 'OK' button to the right. The final section, 'Настройки оболочки Мандельброта' (Mandelbrot Shell Settings), contains three fields: 'Степень' (Degree) set to 8.0, 'Радиус отсечения' (Truncation radius) set to 3.0, and 'Количество итераций' (Number of iterations) set to 50.

Настройки цвета	
Цвет объекта	Цвет фона
	

Выбор объекта	
Оболочка Мандельброта	⌵

Выбор модели рендера	
Марширование лучей (OpenGL)	⌵

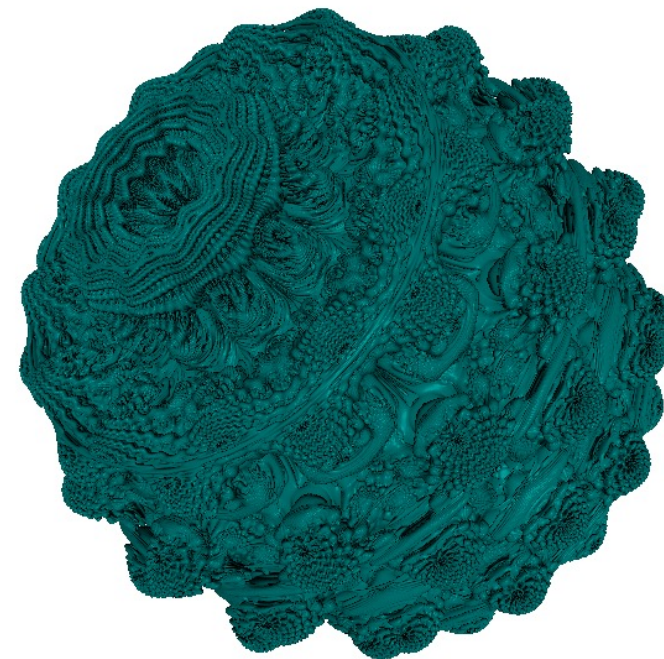
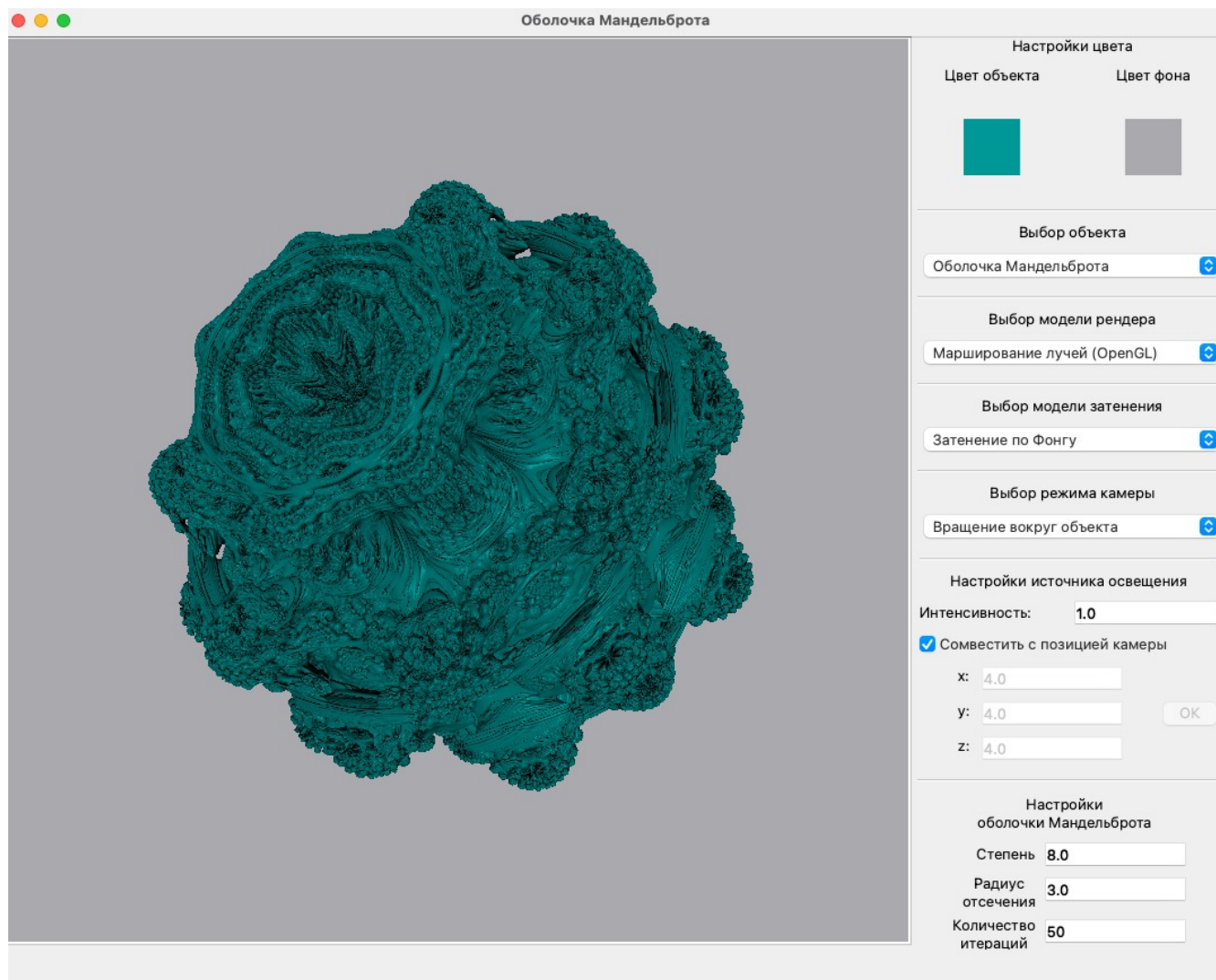
Выбор модели затенения	
Затенение по Фонгу	⌵

Выбор режима камеры	
Вращение вокруг объекта	⌵

Настройки источника освещения	
Интенсивность:	1.0
<input checked="" type="checkbox"/> Совместить с позицией камеры	
x:	4.0
y:	4.0
z:	4.0
OK	

Настройки оболочки Мандельброта	
Степень	8.0
Радиус отсечения	3.0
Количество итераций	50

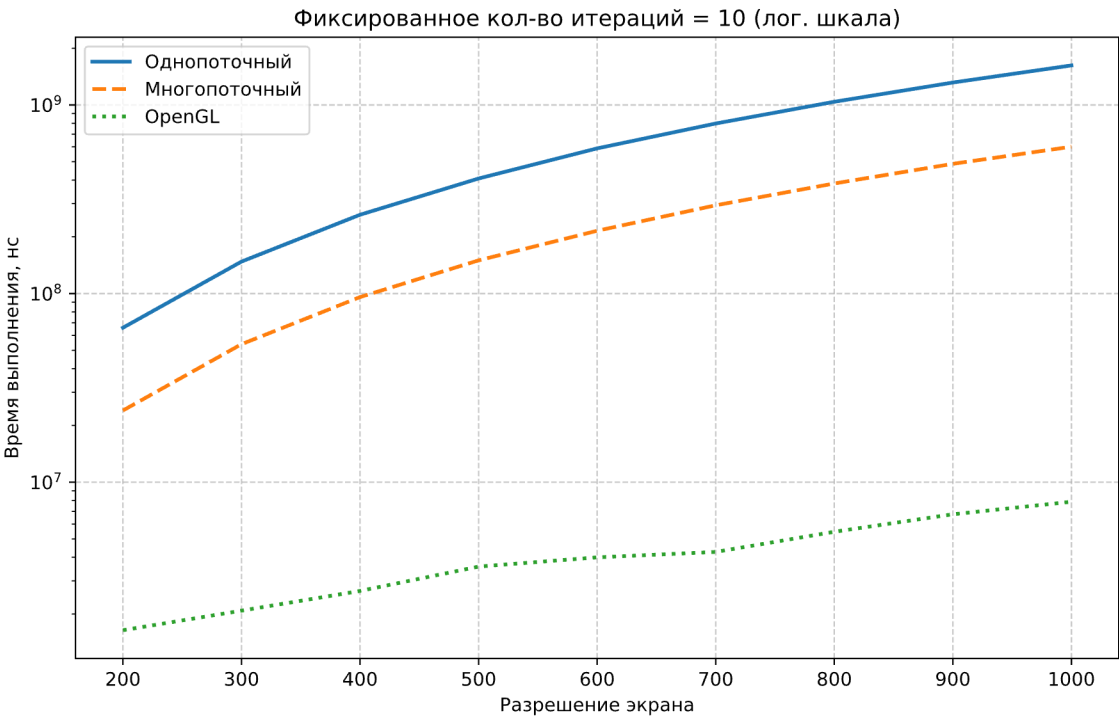
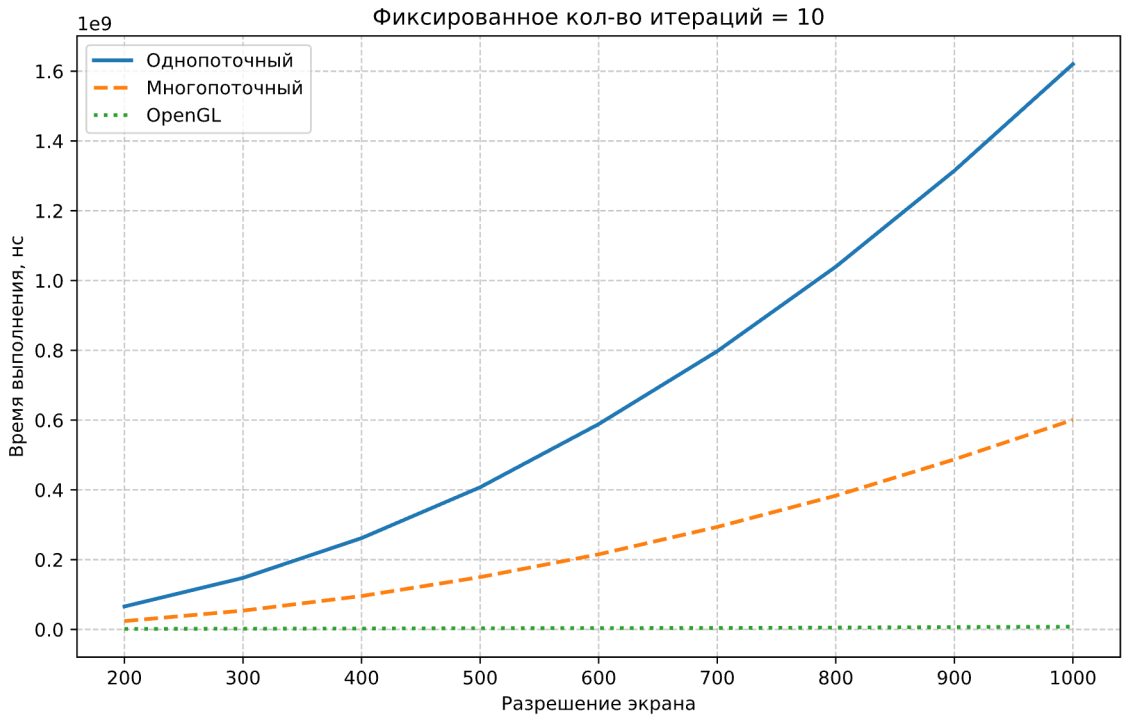
Результат работы программы



Исследование влияния параметров сцены на производительность программы (1)

Исследование зависимости
производительности
программы, как времени
отрисовки одного кадра (в нс)
от **размеров сцены**.

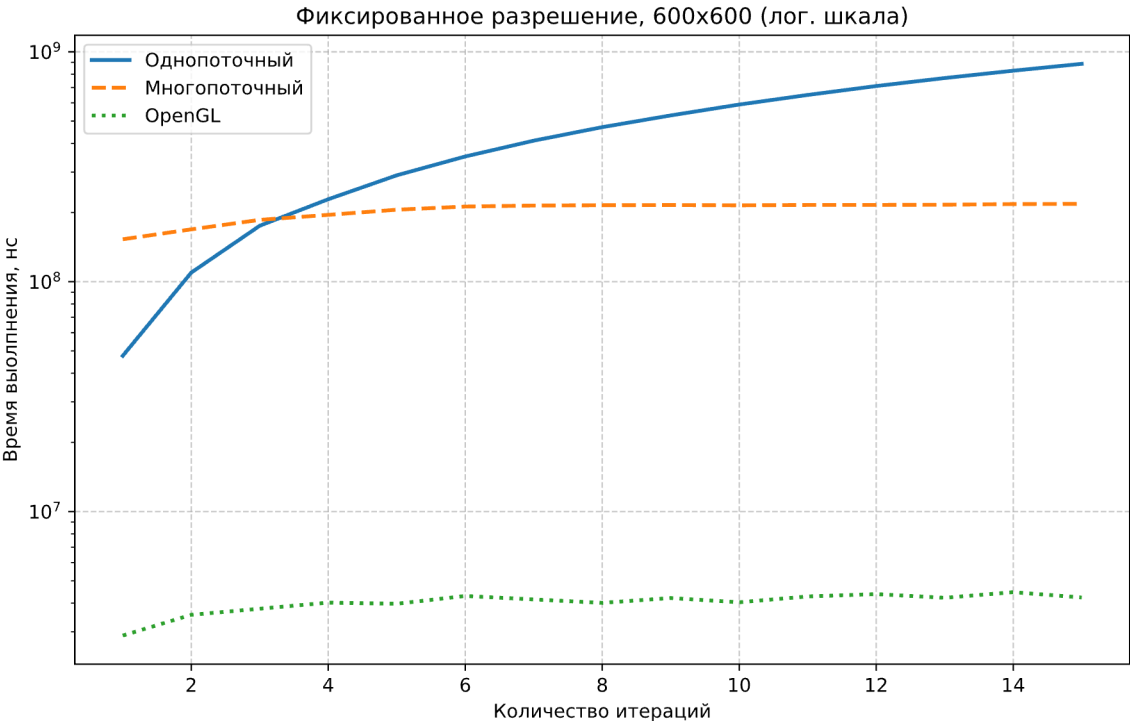
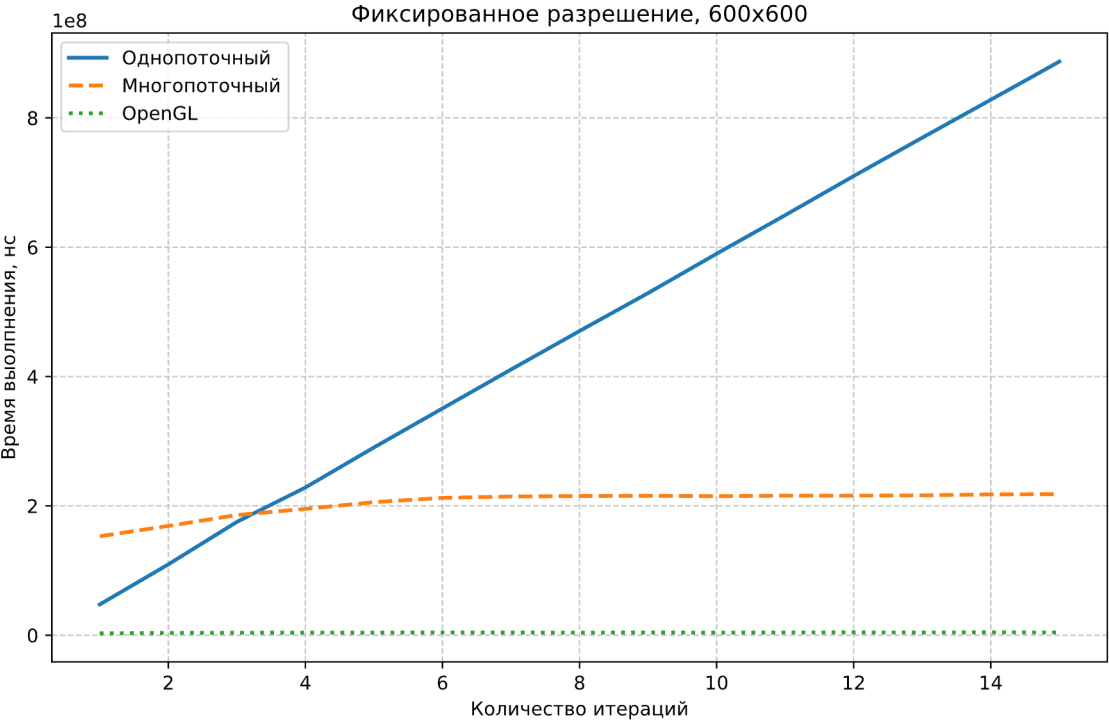
Разрешение	Однопоточный	Многопоточный	OpenGL
200	65895366	23999063	1641956
300	147749941	53905087	2084237
400	261793954	95804413	2651755
500	407500025	149992919	3569735
600	588554860	215305649	3995359
700	797428490	293810629	4269103
800	1039635932	383417703	5459058
900	1314811132	487414066	6760236
1000	1620256803	600612492	7875581



Исследование влияния параметров сцены на производительность программы (2)

Исследование зависимости
производительности
программы, как времени
отрисовки одного кадра (в нс)
от количества итераций
функции SDF для нахождения
расстояния от точки до
фрактала.

Кол-во итераций	Однопоточный	Многопоточный	OpenGL
1	47593287	153015446	2887503
2	109548455	168974550	3554830
3	175185833	185803137	3778188
4	228460870	195374127	4016154
5	290343735	205713838	3971579
6	350660671	212353142	4293275
7	410860620	214479487	4147245
8	470393082	215181351	4007472
9	529079199	215591820	4205380
10	589931072	214973965	4030680
11	649500219	215758280	4273324
12	710061214	215801329	4374386
13	769308330	216255990	4215994
14	827978309	217695584	4459785
15	886998022	218102497	4228779



Заключение

Цель данной курсовой работы была достигнута. Была разработана программа для моделирования изображения трехмерного фрактала «оболочка Мандельброта».

Все поставленные задачи решены.

- проанализированы и выбраны методы и алгоритмы для визуализации трехмерных фракталов;
- реализованы алгоритмы рендера сцены, закраски объектов и нахождения расстояния до них;
- разработан пользовательский интерфейс для управления параметрами сцены, изображаемого объекта и положением камеры;
- проведено исследование производительности программы при разных параметрах сцены; эффективность работы программы зависит от сложности изображаемого объекта и разрешения экрана.