



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

Презентация к курсовой работе на тему

# Генератор трехмерного ландшафта

Дисциплина: Компьютерная графика  
Студент: Лысцев Никита Дмитриевич ИУ7-53Б  
Научный руководитель: Филиппов Михаил Владимирович

Москва, 2023 г.

# Цель и задачи

Цель работы – разработка программного обеспечения для генерации и визуализации трехмерного ландшафта.

Для достижения поставленной цели необходимо решить следующие задачи:

- 1) выполнить формализацию объектов синтезируемой сцены;
- 2) провести анализ существующих алгоритмов создания ландшафта и визуализации сцены, выбрать подходящие и обосновать их выбор;
- 3) разработать ПО;
- 4) реализовать выбранные алгоритмы;
- 5) провести исследование временных характеристик разработанного ПО.

# Формализация объектов синтезируемой сцены

Сцена состоит из следующих объектов:

- ландшафт — трехмерная модель, описываемая полигональной сеткой;
- источник света — материальная точка, испускающая лучи света.

# Выбор алгоритмов (1)

Таблица 1: Сравнение способов представления данных о ландшафте

| Способ                   | Наглядность представления данных | Сложность модификации данных |
|--------------------------|----------------------------------|------------------------------|
| Регулярная сетка         | Высокая                          | Низкая                       |
| Иррегулярная сетка       | Высокая                          | Средняя                      |
| Посегментная карта высот | Средняя                          | Высокая                      |

Выбор: регулярная сетка

Таблица 2: Сравнение алгоритмов процедурной генерации ландшафта

| Алгоритм          | Качество ландшафта | Отсутствие артефактов | Контроль ландшафта |
|-------------------|--------------------|-----------------------|--------------------|
| Diamond-Square    | Среднее            | -                     | Низкий             |
| Холмовой алгоритм | Среднее            | +                     | Средний            |
| Шум Перлина       | Высокое            | +                     | Высокий            |

Выбор: шум Перлина

# Выбор алгоритмов (2)

Таблица 3: Сравнение алгоритмов удаления невидимых линий и поверхностей

| Алгоритм                              | Сложность алгоритма | Скорость работы | Типы объектов          |
|---------------------------------------|---------------------|-----------------|------------------------|
| Алгоритм Робертса                     | $O(n^2)$            | Средняя         | Выпуклые многогранники |
| Алгоритм с z-буфером                  | $O(np)$             | Высокая         | Произвольные           |
| Алгоритм с обратной трассировки лучей | $O(np)$             | Низкая          | Произвольные           |

Выбор: алгоритм с Z-буфером

Таблица 3: Сравнение моделей освещения

| Модель освещения | Реалистичность изображения | Объем вычислений |
|------------------|----------------------------|------------------|
| Модель Ламберта  | Высокая                    | Низкая           |
| Модель Фонга     | Высокая                    | Большой          |

Выбор: модель Ламберта

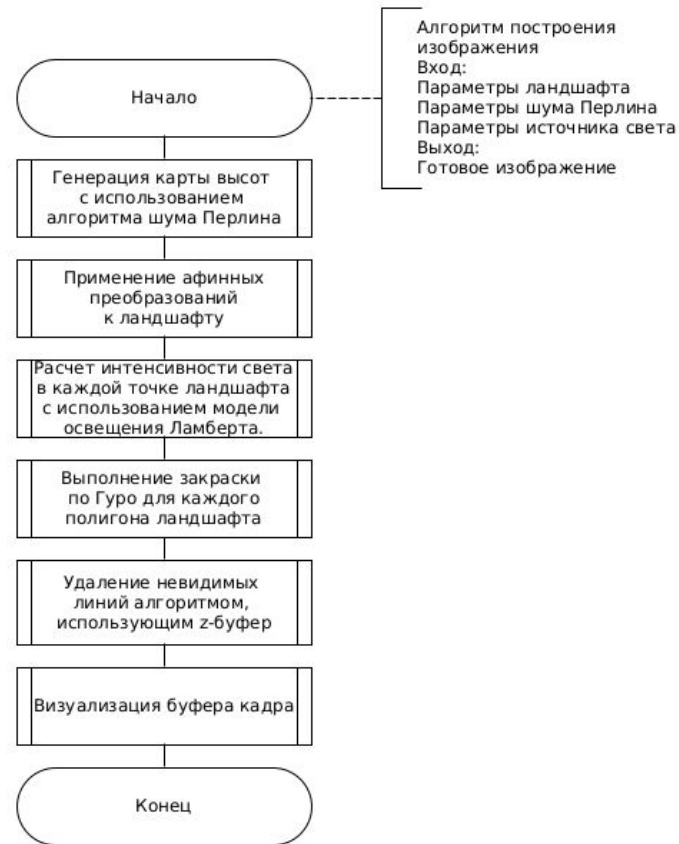
# Выбор алгоритмов (3)

Таблица 3: Сравнение алгоритмов закрашки

| Алгоритм          | Скорость работы | Реалистичность изображения | Сочетание с диффузным отражением |
|-------------------|-----------------|----------------------------|----------------------------------|
| Простая закрашка  | Высокая         | Низкая                     | Высокое                          |
| Закраска по Гуро  | Средняя         | Средняя                    | Высокое                          |
| Закраска по Фонгу | Низкая          | Высокая                    | Средняя                          |

Выбор: закрашка по Гуро

# Схема построения одного кадра изображения



# Структура ПО

В разрабатываемом программном обеспечении реализуются следующие классы:

- PerlinNoise — класс, хранящий параметры алгоритма шума Перлина и реализующий возможность генерации высоты для переданной точки;
- Plane — класс для представления плоскости, являющейся треугольным полигоном;
- Transform — класс для осуществления аффинных преобразований;
- Light — класс для представления точечного источника света;
- LightManager — класс для вычисления интенсивностей света в точке;
- Landscape — класс для представления трехмерного ландшафта;
- LandscapeManager — класс для осуществления всех операций по изменению ландшафта;
- Renderer — класс для растеризации ландшафта и вывода его на экран.



# Средства реализации

Для реализации программного продукта был выбран C++ по следующим причинам:

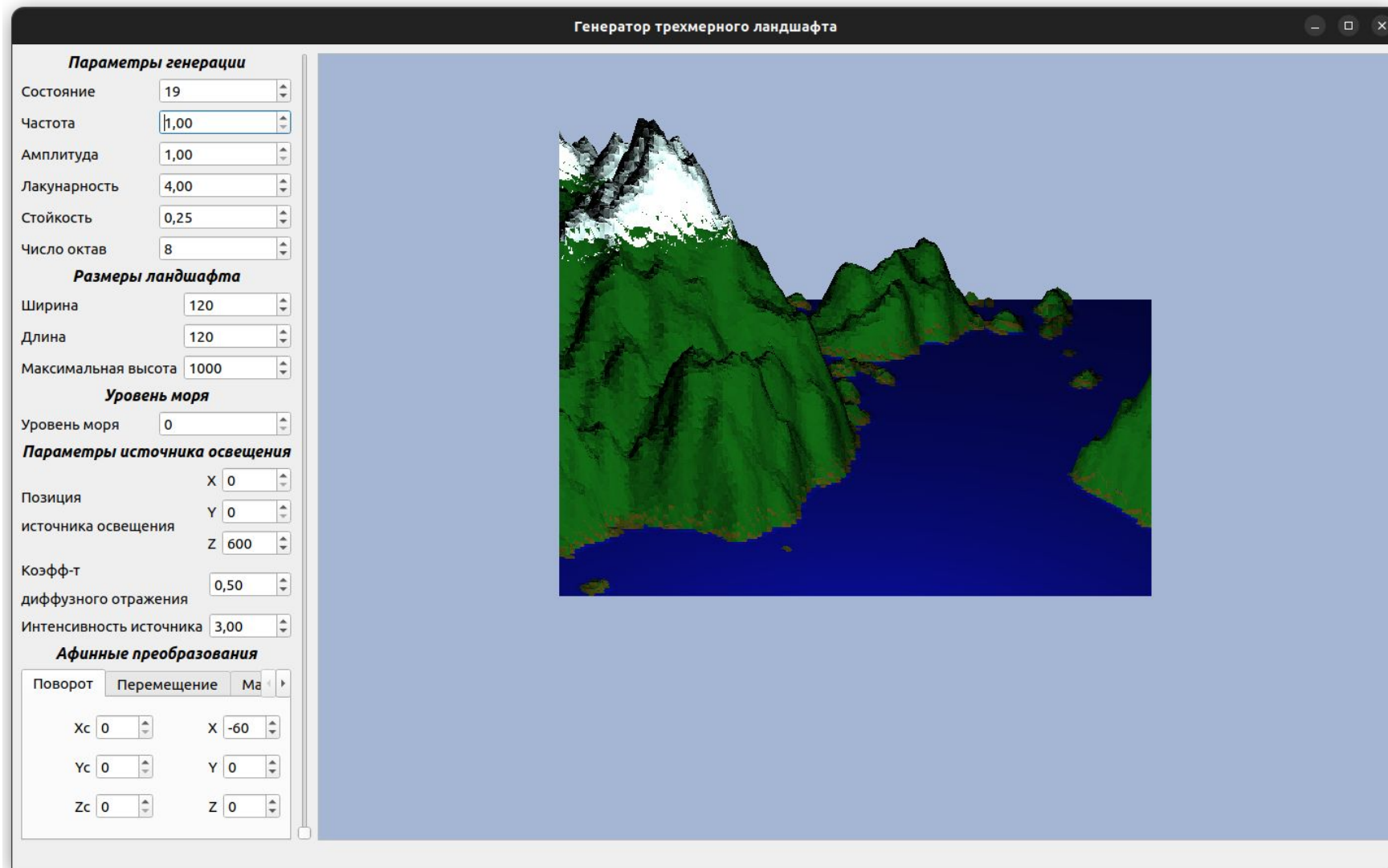
- В стандартной библиотеке языка присутствует поддержка всех структур данных, выбранных по результатам проектирования;
- средствами C++ можно реализовать все алгоритмы, выбранные в результате проектирования.

В качестве среды разработки предпочтение было отдано среде QT Creator по следующим причинам:

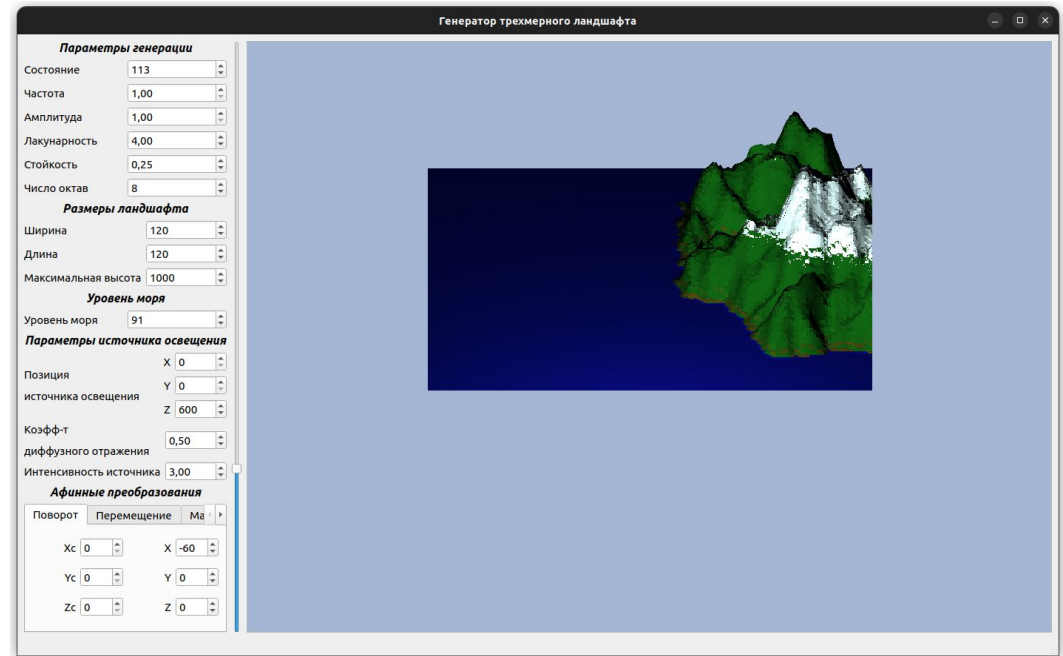
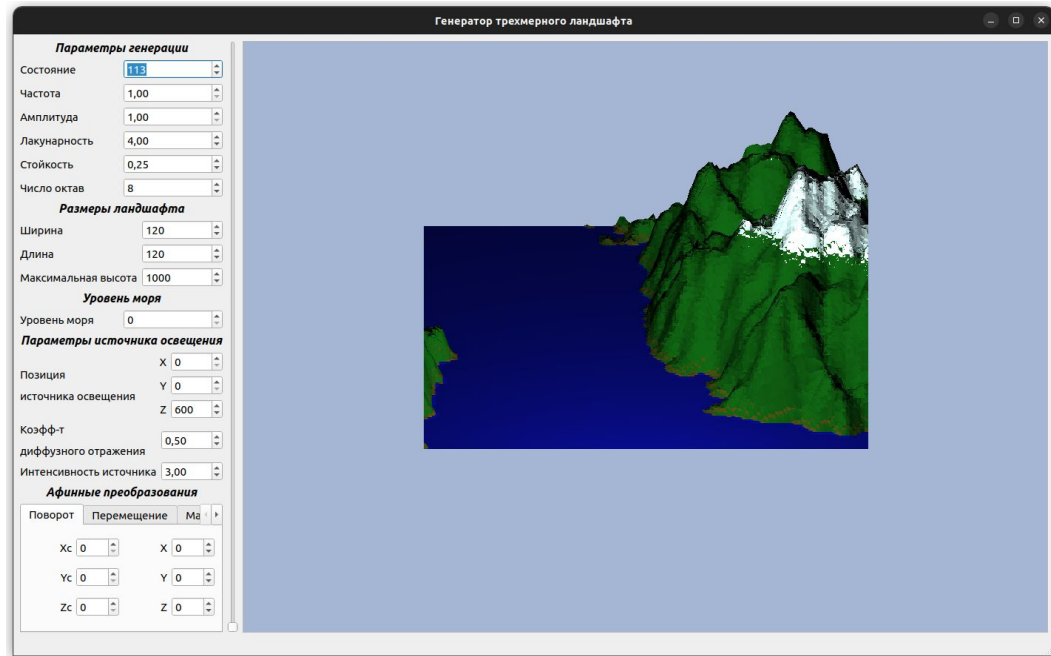
- данная среда поставляется с фреймворком Qt, который содержит в себе все необходимые средства, позволяющие работать непосредственно с пикселями изображения;
- QT Creator позволяет работать с расширением QT Design, который позволяет создавать удобный и надежный интерфейс.

Для упрощения и автоматизации сборки проекта используется утилита cmake.

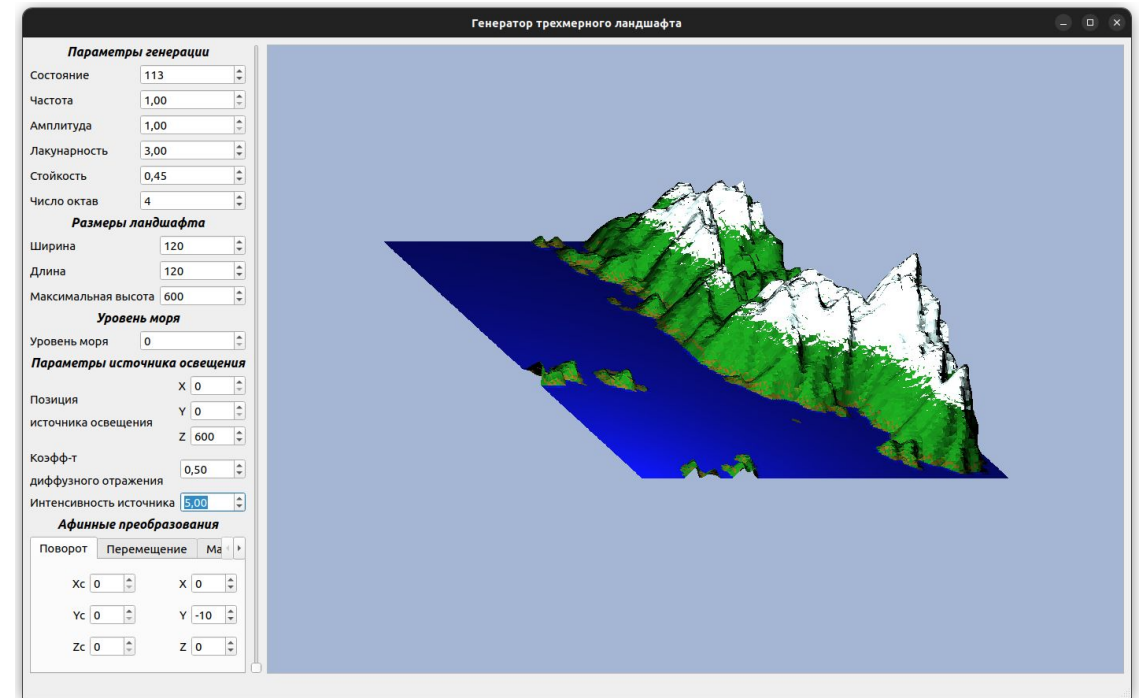
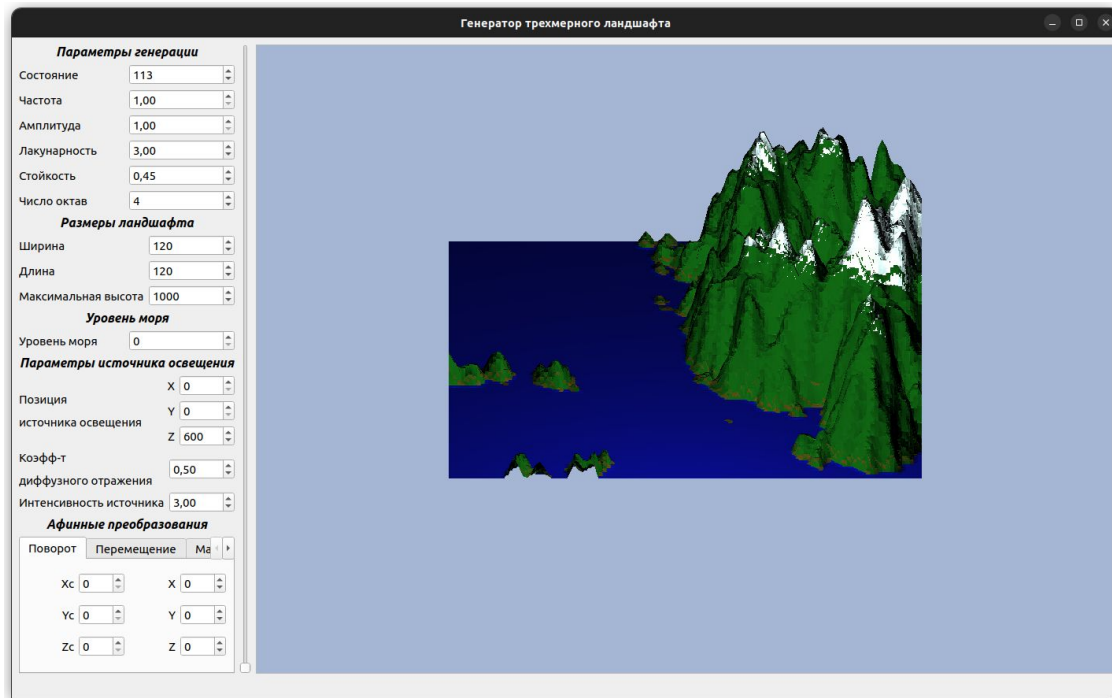
# Интерфейс программы



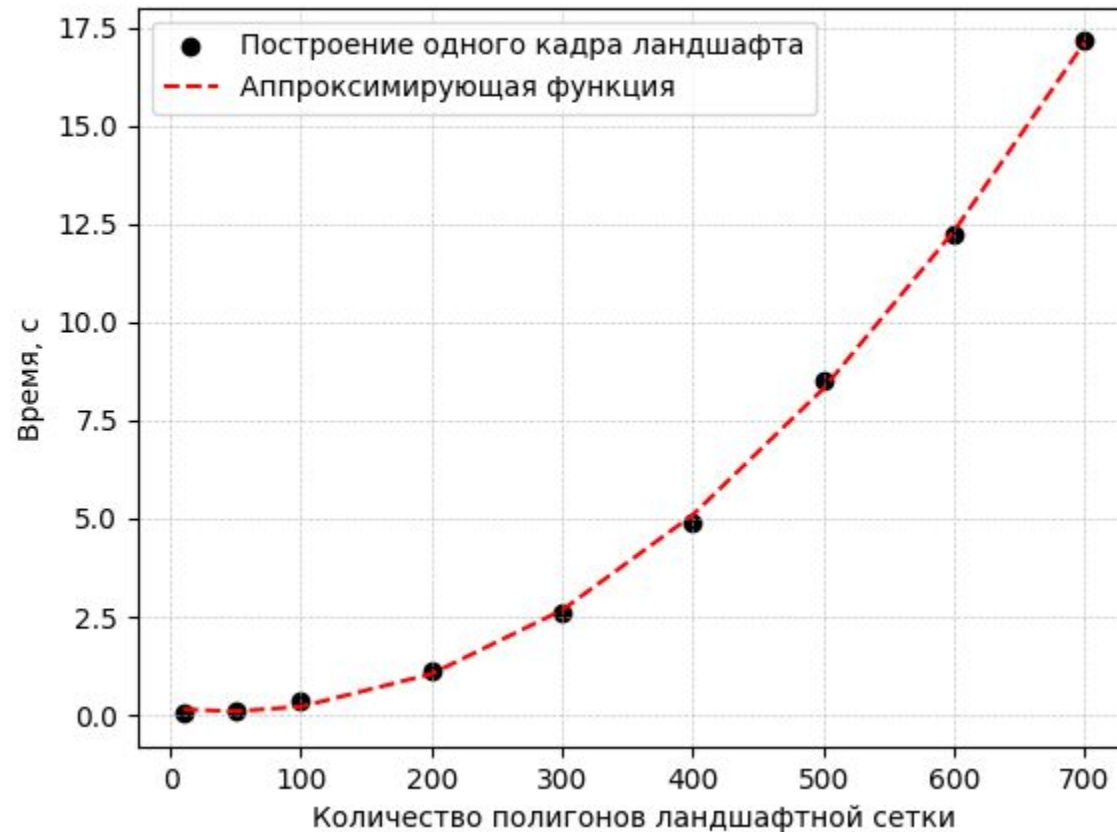
# Примеры работы программы(1)



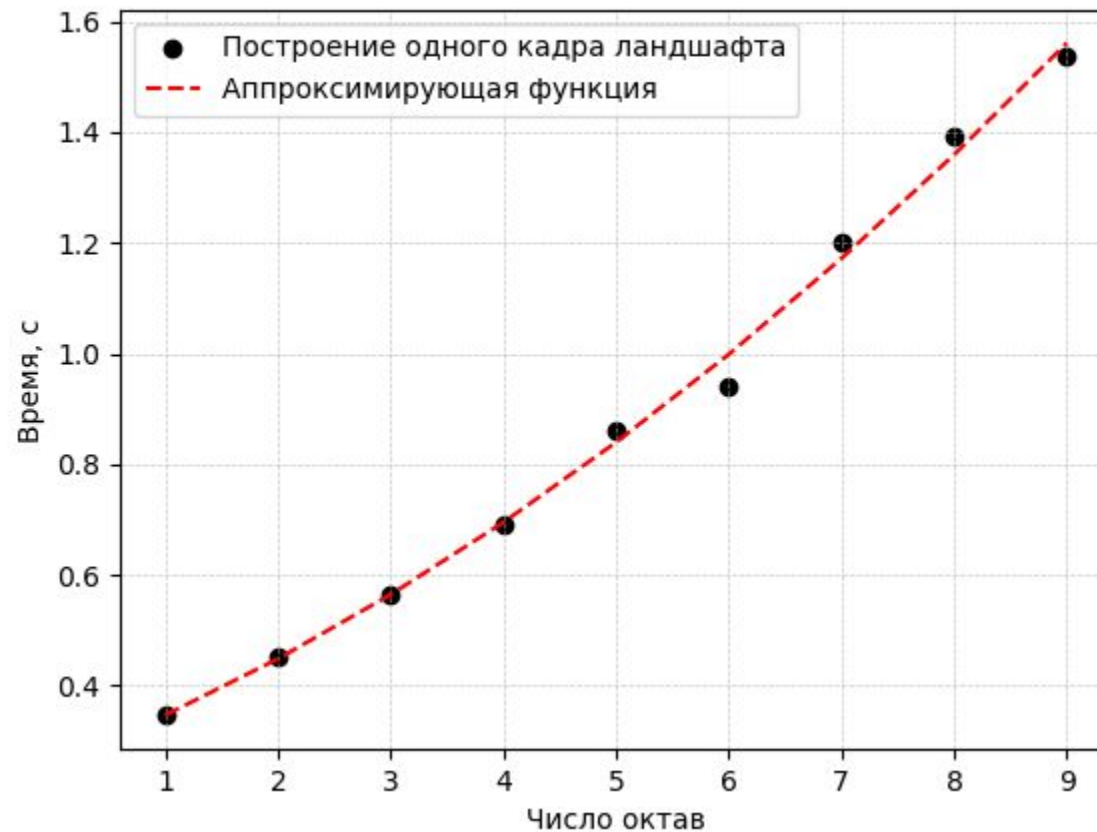
# Примеры работы программы(2)



# Зависимость времени построения одного кадра ландшафта от размеров линейных размеров ландшафта



# Зависимость времени построения одного кадра ландшафта от числа октав



# Заключение

В ходе выполнения курсового проекта были решены следующие задачи:

- 1) выполнена формализация объектов синтезируемой сцены;
- 2) проведен анализ существующих алгоритмов создания ландшафта и визуализации сцены, выбраны подходящие и обоснован их выбор;
- 3) разработано ПО;
- 4) реализованы выбранные алгоритмы;
- 5) проведено исследование временных характеристик разработанного ПО.

Цель работы, а именно разработка программного обеспечения для генерации и визуализации трехмерного ландшафта также была достигнута.