

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук
Департамент программной инженерии

СОГЛАСОВАНО

Доцент департамента больших данных и
информационного поиска факультета
компьютерных наук.

_____ Е. О. Кантонистовая
«__» _____ 2024 г.

УТВЕРЖДАЮ

Академический руководитель
ОП “Программная инженерия”

_____ Н. А. Павлочев
«__» _____ 2024 г.

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

3D RENDERER С НУЛЯ
Программа и методика испытаний

ЛИСТ УТВЕРЖДЕНИЯ
RU.17701729.05.01-01 51 01-1 ЛУ

Исполнитель:
студент группы БПИ223
_____ / Гетманов М. М. /
«__» _____ 2024 г

Москва 2024

УТВЕРЖДЕН

RU.17701729.05.01-01 51 01-1

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл.	

3D RENDERER С НУЛЯ

Программа и методика испытаний

RU.17701729.05.01-01 51 01-1

Листов 24

Москва 2024

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

АННОТАЦИЯ

Программа и методика испытаний — это документ, в котором содержится информация о программном продукте, а также полное описание приемочных испытаний для данного программного продукта.

Настоящая Программа и методика испытаний для «3D Renderer» содержит следующие разделы: «Объект испытаний», «Цель испытаний», «Требования к программе», «Требования к программной документации», «Средства и порядок испытаний», «Методы испытаний», «Приложения».

В разделе «Объект испытаний» указано наименование, краткая характеристика и назначение программы.

В разделе «Цель испытаний» указана цель проведения испытаний.

Раздел «Требования к программе» содержит основные требования к программе, которые подлежат проверке во время испытаний (требования к функционалу и интерфейсу).

Раздел «Требования к программным документам» содержит состав программной документации, которая представляется на испытания.

Раздел «Средства и порядок испытаний» содержит информацию о технических и программных средствах, которые следует использовать во время испытаний, а также порядок этих испытаний.

Раздел «Методы испытаний» содержит информацию об используемых методах испытаний.

Настоящий документ разработан в соответствии с требованиями:

- 1) ГОСТ 19.101-77 Виды программ и программных документов [1];
- 2) ГОСТ 19.102-77 Стадии разработки [2];
- 3) ГОСТ 19.103-77 Обозначения программ и программных документов [3];
- 4) ГОСТ 19.104-78 Основные надписи [4];
- 5) ГОСТ 19.105-78 Общие требования к программным документам [5];
- 6) ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом [6];

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

7) ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению [7].

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

СОДЕРЖАНИЕ

АННОТАЦИЯ	2
1. ОБЪЕКТ ИСПЫТАНИЙ	5
1.1. Наименование программы.....	5
1.2. Краткая характеристика и область назначения	5
2. ЦЕЛЬ ИСПЫТАНИЙ	6
3. ТРЕБОВАНИЯ К ПРОГРАММЕ	7
3.1. Требования к техническим характеристикам	7
3.2. Требования к входным и выходным данным	8
3.3. Требования к интерфейсу.....	10
4. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ	13
4.1. Состав программной документации	13
4.2. Специальные требования к программной документации.....	13
5. СРЕДСТВА И ПОРЯДОК ИСПЫТАНИЙ	14
5.1. Технические средства.....	14
5.2. Программные средства	14
5.3. Порядок проведения испытаний	14
5.4. Требования к персоналу	14
6. МЕТОДЫ ИСПЫТАНИЙ.....	15
6.1. Проверка требований к технической документации.....	15
6.2. Проверка требований к интерфейсу	15
6.3. Проверка требований к функциональным характеристикам.....	15
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ	20
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	22

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

1. ОБЪЕКТ ИСПЫТАНИЙ

1.1. Наименование программы

Наименование программы – «3D Renderer».

Наименование программы на английском языке – «3D Renderer».

Краткое наименование программы, используемое далее в документе – Приложение.

1.2. Краткая характеристика и область назначения

«3D Renderer» - интерактивный образовательный проект. Его цель – научиться имплементировать последовательность отрисовки 3D объектов.

Интерактивность Приложения заключается в том, что есть возможность не только наблюдать за статической картинкой, на которой изображены 3D объекты, но и перемещать камеру, через которую просматриваются объекты, в пространстве.

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

2. ЦЕЛЬ ИСПЫТАНИЙ

Целью испытаний является проверка корректности выполнения программой функций, изложенных в п. 4 «Требования к программе» документа «Техническое задание» из комплекта документации в соответствии с ЕСПД (Единой системой программной документации)

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

3. ТРЕБОВАНИЯ К ПРОГРАММЕ

Программа должна соответствовать следующим функциональным требованиям, указанным в документе «Мобильное приложение для онлайн соревнования в решении кроссвордов. Техническое задание»:

3.1. Требования к техническим характеристикам

3.1.1. Состав выполняемых функций:

1. Демонстрационное приложение.

Запуск программы осуществляется с помощью запуска соответствующего исполняемого файла, генерируемого системой сборки проекта.

При запуске программы должно появиться два окна – окно 1 и окно 2(консоль) (окна описаны в пункте 4.3). Окно 1 должно отображать текущее положение 3D сцены, которое видит камера. Объекты добавляются на сцену пользователем через консоль.

Приложение может находиться в двух состояниях:

- a. Отрисовка сцены в окне 1. В этом состоянии пользователь может управлять положением камеры на сцене. В консоли появляется информация о работе приложения в соответствии с пунктом 4.1.3.
- b. Добавление объектов на сцену. В этом состоянии отрисовка сцены в окне 1 приостановлена для добавления нового объекта. В консоли пользователь должен ввести информацию о новом объекте в формате, указанном в пункте 4.1.2. При несоответствии форматов должна выводиться информация в соответствии с пунктом 4.1.3.

Переход между двумя состояниями обрабатывается следующим образом:

1. Из состояния “a” в состояние “b” пользователь переходит с помощью нажатия клавиши “P”
2. Из состояния “b” в состояние “a” пользователь переходит с помощью ввода команды “exit” на любом этапе ввода информации о новом объекте или при завершении ввода информации о новом объекте или при неверном вводе информации о новом объекте.

Действия пользователя в состоянии “a” обрабатываются с помощью run-time loop.

Завершение работы программы обеспечивается встроенными средствами операционной системы (нажатие кнопки “заккрыть” у окна программы).

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

2. Библиотека отрисовки.

Библиотека содержит алгоритмы отрисовки объектов на 3D сцене. Библиотека должна поддерживать отрисовку объектов, которые представляются в виде набора треугольников. Также библиотека должна уметь работать с фоновым и направленным параллельным светом.

В библиотеке должны находиться следующие классы:

1. World, который содержит в себе информацию о глобальной системе координат, находящихся на ней объектах, источниках света и камеры.
2. Renderer, который отрисовывает на пиксельном экране состояние объекта World, которое видит камера.
3. Camera, через которую пользователь может видеть текущее состояние сцены.
4. TriangulatedObject, который описывает в пространстве 3D объект, представимый в виде набора треугольников.

3.2. Требования к входным и выходным данным

3.2.1. Организация входных данных:

В состоянии “a”:

1. Пользователь с помощью клавиш WASD управляет местоположением камеры на сцене. Перемещение камеры происходит в системе координат камеры.
2. Пользователь с помощью стрелок на клавиатуре управляет наклонами камеры. Наклон камеры осуществляется в системе координат камеры.
3. Пользователь с помощью клавиши “P” переходит в состояние “b”.

В состоянии “b”:

1. Пользователь вводит путь к файлу, содержащий информацию о новом объекте в формате .obj. Содержимое файла должно удовлетворять правилам, находящимся по ссылке: <https://paulbourke.net/dataformats/obj/>. При вводе команды “exit” программа переходит в состояние “a”.
2. После ввода пути к файлу пользователь вводит координаты нового объекта на сцене в формате: “x y z” – каждая координата является вещественным числом, записанным с помощью цифр и точки в качестве знака разделителя целой части и дробной. Координаты идут через пробел. Первое число отвечает за координаты по оси “x”, второе число отвечает за координаты по оси “y”, третье число отвечает за

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

координаты по оси “z”. При вводе команды “exit” вместо первого числа программа переходит в состояние “a”.

- После ввода координат объекта пользователь вводит цвет объекта в формате: “R G B” – три целых числа, идущие подряд, каждое от 0 до 255 включительно, первое число отвечает за красную составляющую в цвете, второе число отвечает за зелёную составляющую в цвете, третье число отвечает за синюю составляющую в цвете. При вводе команды “exit” вместо первого числа программа переходит в состояние “a”.

3.2.2. Организация выходных данных:

В состоянии “a”:

- В окно 1 выводится картинка, соответствующая текущему состоянию сцены и тому, что видит камера.
- В консоль выводится информация о том, за сколько был отрисован очередной кадр. Формат: “New frame was rendered in: <число> ms”, где <число> - количество миллисекунд, за которое был отрисован очередной кадр.
- Любая другая дополнительная информация является опциональной для записи в консоль.

В состоянии “b”:

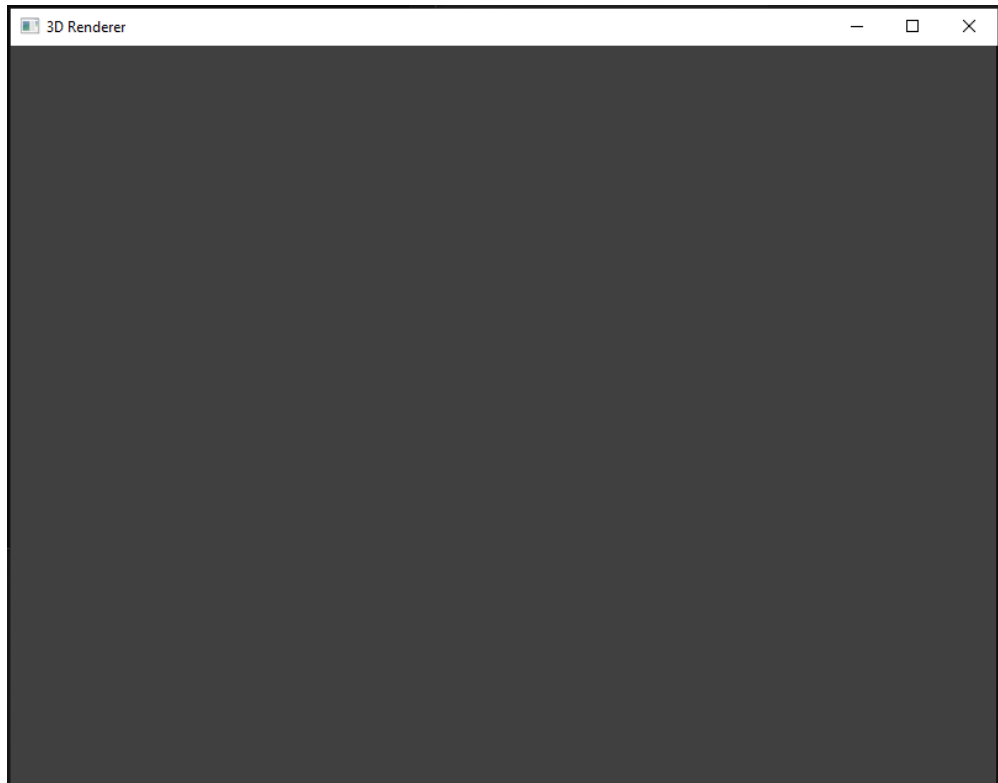
- При переходе в текущее состояние в консоли должна появиться надпись о том, что отрисовка сцены остановлена и для перехода в состояние “a” необходимо ввести команду “exit”.
- При вводе пути к файлу с новым объектом в консоль выводится подсказка для пользователя о том, что сейчас необходимо ввести путь к файлу в формате .obj.
- При вводе координат нового объекта в консоль выводится подсказка для пользователя о том, что сейчас необходимо ввести координаты объекта, а также информация о формате ввода данных.
- При вводе цвета нового объекта в консоль выводится подсказка для пользователя о том, что сейчас необходимо ввести цвет объекта, а также информация о формате ввода данных.
- Если в пунктах 2–4 происходит ошибка формата вводимых данных, должно появиться сообщение о несоответствии введённых данных ожидаемому формату.

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

3.3. Требования к интерфейсу

Приложение имеет два окна:

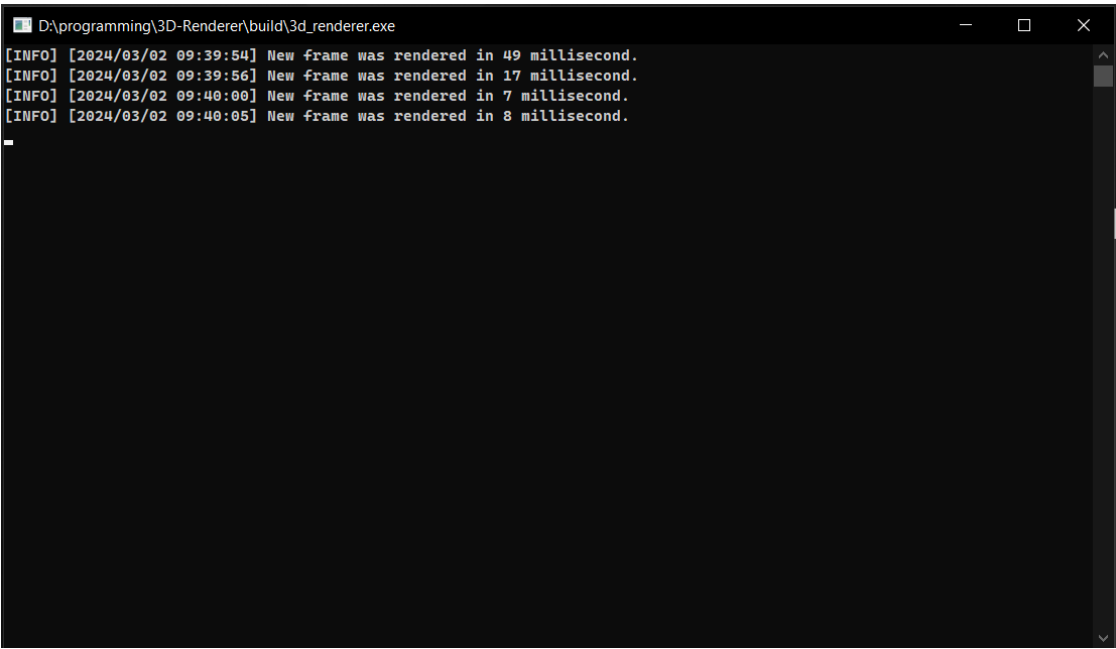
1. Окно 1 для отрисовки 3D сцены:



Окно полностью черное, так как на сцене ничего нет.

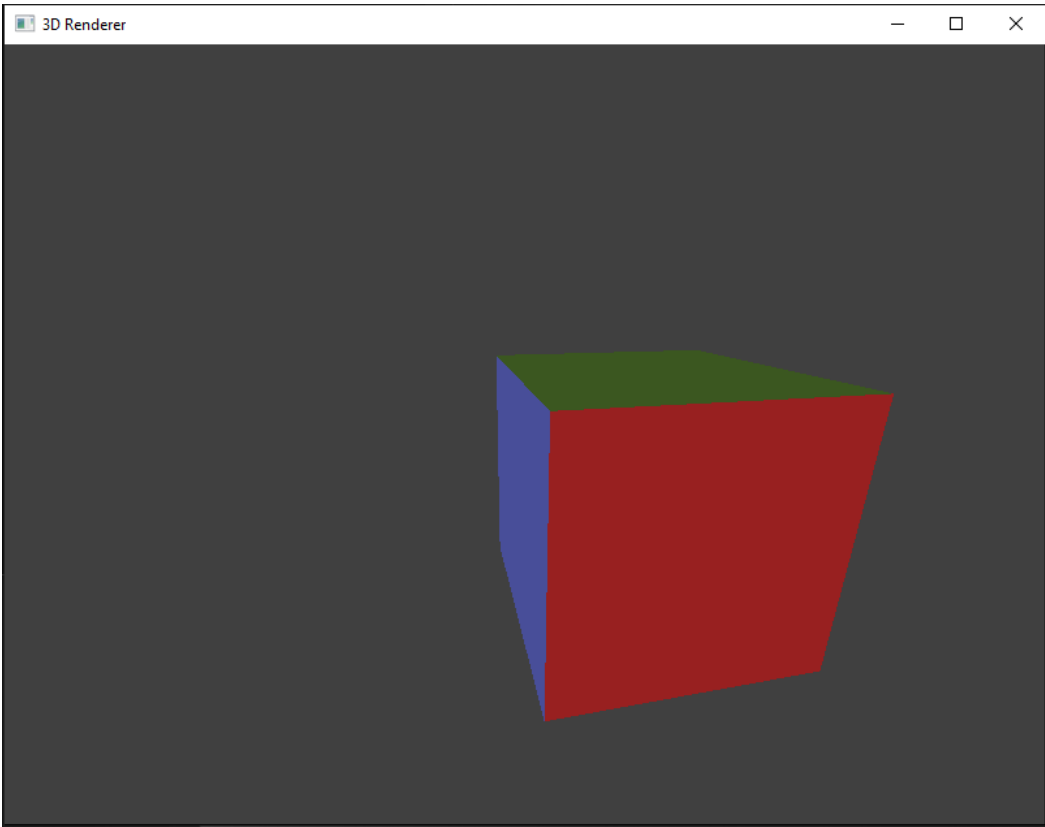
2. Окно 2 - Консоль для управления наполнением сцены и вывода дополнительной информации:

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата



3. Примеры состояния окон:

3.1. На сцене находится куб



3.2. Пример успешного ввода данных о новом объекте:

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

```
D:\programming\3D-Renderer\build\3d_renderer.exe
[INFO] [2024/03/02 09:39:54] New frame was rendered in 49 millisecond.
[INFO] [2024/03/02 09:39:56] New frame was rendered in 17 millisecond.
[INFO] [2024/03/02 09:40:00] New frame was rendered in 7 millisecond.
[INFO] [2024/03/02 09:40:05] New frame was rendered in 8 millisecond.
[INFO] [2024/03/02 09:41:09] New frame was rendered in 7 millisecond.
[INFO] Enter path to .obj file with new object:
D:\programming\3D-Renderer\object_samples\cube.obj
[INFO] Enter color of new object: (format: <R> <G> <B>, where R, G, B are numbers, each number is integer and >=0 and <255)
[INFO] Example: 128 0 255
128 128 128
[INFO] Enter position of new object: (format: <x> <y> <z>, where x, y, z are real numbers)
[INFO] Example: 1.5 -0.5 0
0 0 -3
[INFO] New object was added successfully!
[INFO] [2024/03/02 09:41:21] New frame was rendered in 90 millisecond.
```

3.3. Сообщение о времени, затраченном на отрисовку сцены:

```
D:\programming\3D-Renderer\build\3d_renderer.exe
[INFO] [2024/02/31 15:40:37] New frame was rendered in 52 millisecond.
```

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

4. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

4.1. Состав программной документации

- «3D Renderer с нуля». Техническое задание (ГОСТ 19.201-78);
- «3D Renderer с нуля». Программа и методика испытаний (ГОСТ 19.30178);
- «3D Renderer с нуля». Текст программы (ГОСТ 19.40178);
- «3D Renderer с нуля». Пояснительная записка (ГОСТ 19.40479);
- «3D Renderer с нуля». Руководство оператора (ГОСТ 19.50579);

4.2. Специальные требования к программной документации

- Документы к программе должны быть выполнены в соответствии с ГОСТ 19.106-78 и ГОСТами к каждому виду документа (см. п. 5.1.);

Документация и программа сдается в электронном виде в формате pdf;

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

5. СРЕДСТВА И ПОРЯДОК ИСПЫТАНИЙ

5.1. Технические средства

Для нормального функционирования программы требуется персональный компьютер, оснащенный следующими техническими компонентами:

1. Для Windows:

Дисплей: Минимальное разрешение 1280 x 720 пикселей.

Операционная система: Windows 10 или более новые версии.

Оперативная память: 4 ГБ или более.

Хранилище: не менее 100 МБ свободного места для хранения.

2. Для Linux:

Дисплей: Минимальное разрешение 1280 x 720 пикселей.

Дистрибутив: Ubuntu 20.04 или более новые версии.

Оперативная память: 4 ГБ или более.

Хранилище: не менее 100 МБ свободного места для хранения.

3. Система сборки CMake

4. Компилятор C и C++

5. Система контроля версий git

5.2. Программные средства

5.3. Порядок проведения испытаний

Испытания должны проводиться в следующем порядке:

1. Проверка требований к программной документации.

2. Проверка требований к интерфейсу.

3. Проверка требований к функциональным характеристикам.

5.4. Требования к персоналу

Минимальное количество персонала, требуемого для работы программы, должно составлять не более 1 человека. Пользователь должен уметь работать с ЯП C++, системой сборки cmake и системой контроля версий git.

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

6. МЕТОДЫ ИСПЫТАНИЙ

6.1. Проверка требований к технической документации

Состав программной документации проверяется наличием всех элементов программной документации в системе SmartLMS. Также проверяется соответствие документации требованиям ГОСТ.

Все документы удовлетворяют представленным требованиям.

6.2. Проверка требований к интерфейсу

Проверка требований к интерфейсу осуществляется в соответствии с пунктом 4.3. документа “Техническое задание”.

Необходимо убедиться в существовании двух окон:

1. Окно 1 – окно для рендеринга.
2. Окно 2 – окно для ввода и вывода информации.

6.3. Проверка требований к функциональным характеристикам

1. Запуск программы

При запуске программы появляется два окна, соответствующих п. 4.3. “Технического задания”.

2. Добавление объекта

При нажатии клавиши Р в окне 2 появляются по очереди три просьбы к пользователю:

- 2.1. Ввод пути к файлу с новым объектом
- 2.2. Ввод цвета нового объекта
- 2.3. Ввод координат нового объекта

Для каждой просьбы должна выводиться подсказка, соответствующая п. 4.1.3.

“Технического задания”. При нарушении формата ввода должно появиться сообщение, соответствующее п. 4.1.3. “Технического задания”.

После корректного ввода данных в окне 2 должно появиться сообщение о том, что новый объект был добавлен.

Пример:

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

При добавлении куба цвета 128 128 128 впереди камеры (отрицательное направление оси oz , например, координаты 0 0 -5) должен появиться красный квадрат посередине окна 1.

3. Проверка управления камерой

После добавления на сцену объекта впереди камеры (рекомендуется использовать готовый файл объекта, а именно cube.obj, из репозитория и поместить его на координаты 0 0 -5, на примере куба будет описана дальнейшая проверка управления камерой) должно происходить следующее:

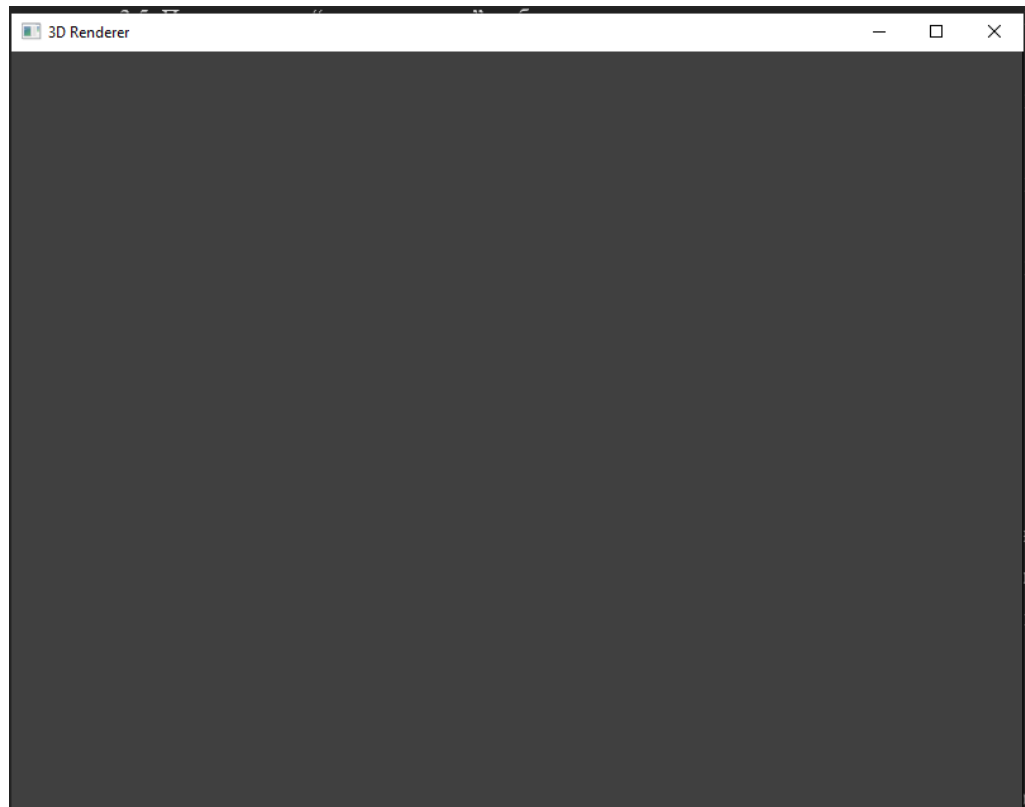
- 3.1. При нажатии клавиши W куб впереди камеры должен стать больше.
 - 3.2. При нажатии клавиши S куб впереди камеры должен стать меньше.
 - 3.3. При нажатии клавиши A куб должен переместиться правее на экране, не меняя своего размера и своей формы.
 - 3.4. При нажатии клавиши D куб должен переместиться левее на экране, не меняя своего размера и своей формы.
- Для выполнения проверки изменения наклонов камеры рекомендуется приближаться к кубу с помощью клавиши W до тех пор, пока его ширина на экране окна 1 не будет примерно равна 1/4 от ширины окна 1.
- 3.5. При нажатии “стрелки вверх” куб должен переместиться ниже на экране окна 1, причём верхняя грань стороны куба должна стать короче, а нижняя - длиннее.
 - 3.6. При нажатии “стрелки вниз” куб должен переместиться выше на экране окна 1, причём верхняя грань стороны куба должна стать длиннее, а нижняя - короче.
 - 3.7. При нажатии “стрелки влево” куб должен переместиться правее на экране окна 1, причём левая грань стороны куба должна стать короче, а правая - длиннее.
 - 3.8. При нажатии “стрелки вправо” куб должен переместиться левее на экране окна 1, причём левая грань стороны куба должна стать длиннее, а правая - короче.

После каждого нажатия клавиши в окне 2 должна появляться информация о том, за какое количество миллисекунд отрисовался новый кадр.

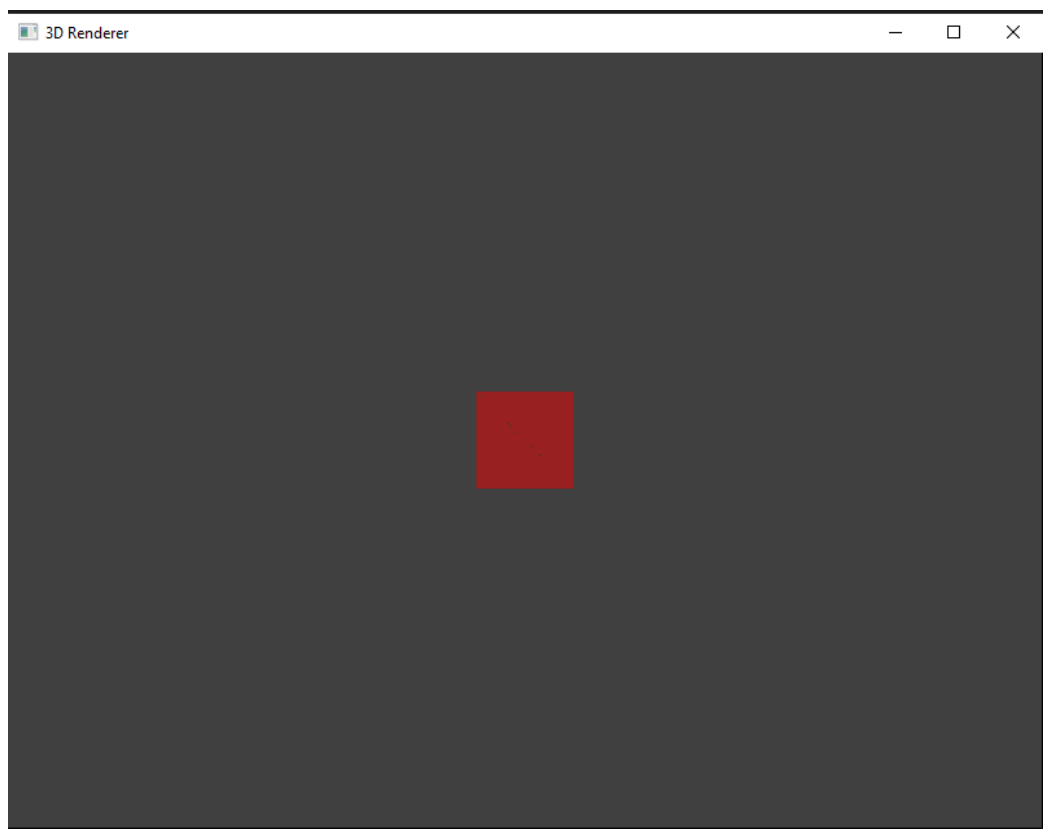
4. Корректность работы света.

При запуске приложения окно 1 должно иметь серый цвет.

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

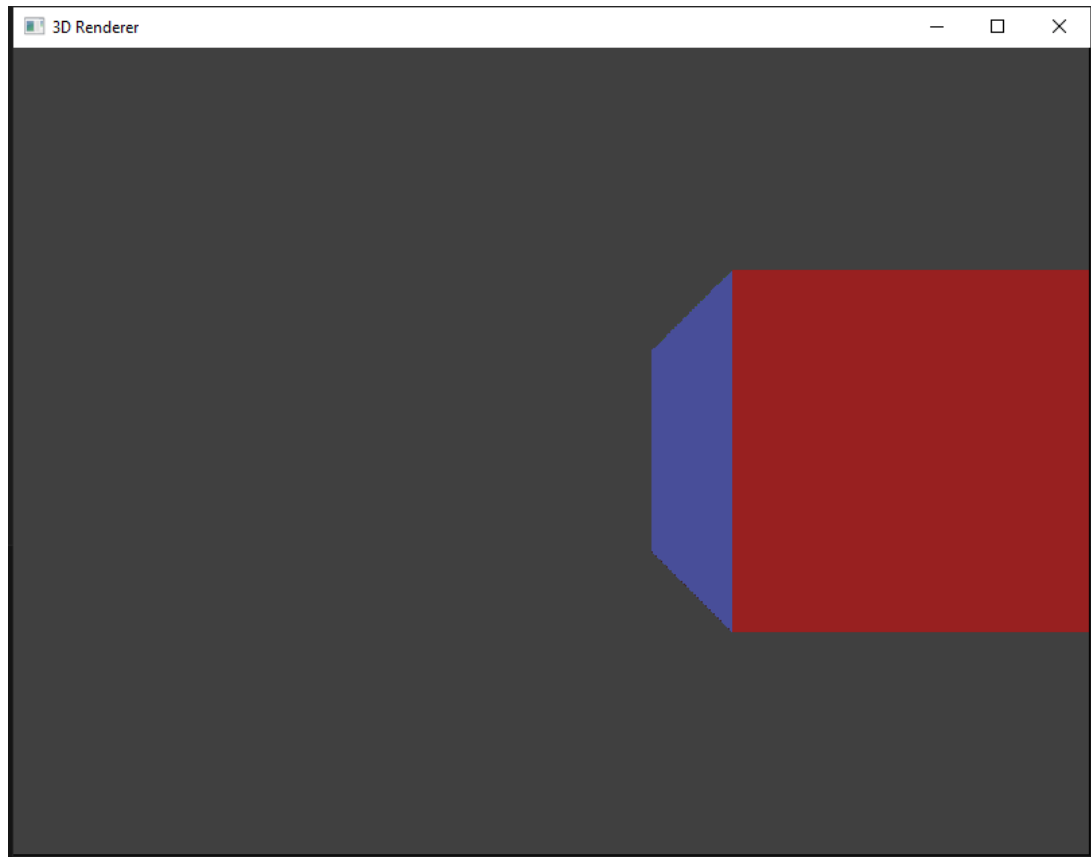


После добавления cube.obj на сцену на координаты 0 0 -5 с цветом 128 128 128 (серый) посередине экрана должен появиться красный квадрат.



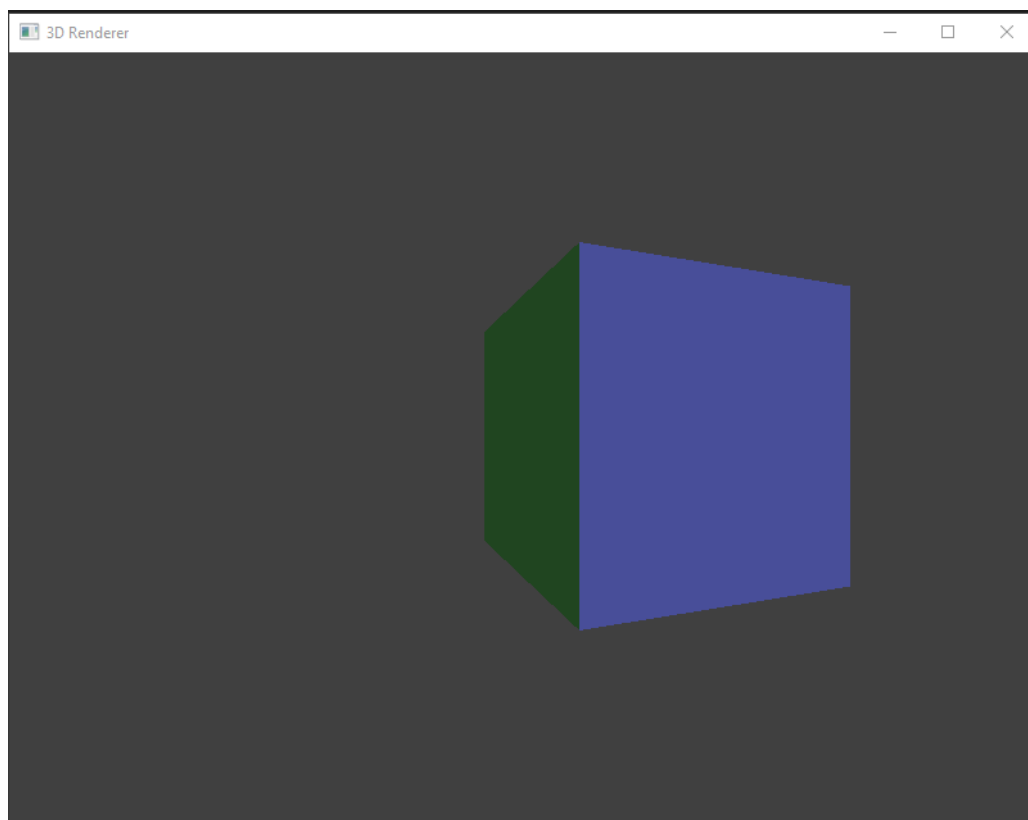
Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

После приближения к нему (клавиша W) и перемещению камеры влево (клавиша A) должна появиться синяя сторона куба.



После перемещения камеры за куб и наведения камеры на куб должна появиться зелёная грань куба.

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата



Вид, размеры куба и его положение на экране окна 1 могут отличаться в зависимости от степени приближения камеры к кубу и угла поворота камеры.

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. ГОСТ 19.101-77 Виды программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
2. ГОСТ 19.102-77 Стадии разработки. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
3. ГОСТ 19.103-77 Обозначения программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
4. ГОСТ 19.104-78 Основные надписи. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
5. ГОСТ 19.105-78 Общие требования к программным документам. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
6. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
7. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
8. ГОСТ 19.603-78 Общие правила внесения изменений. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
9. ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
10. 3-D Computer Graphics, A Mathematical Introduction with OpenGL – Samuel R. Buss, Cambridge 2003
11. Mathematics for 3D Game Programming and Computer Graphics Third Edition – Eric Lengyel, Course Technology PTR 2012
12. Компьютерная графика. Динамика, реалистические изображения – Е.В. Шикин, А. В. Боресков, Москва “Диалог-МИФИ” 1995
13. Компьютерная графика. Полигональные модели – Е.В. Шикин, А. В. Боресков, Москва “Диалог-МИФИ” 2001
14. Eigen. Eigen: A C++ template library for linear algebra. URL: https://eigen.tuxfamily.org/index.php?title=Main_Page
15. SFML. SFML: Simple and Fast Multimedia Library. URL: <https://www.sfml-dev.org/index.php>

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

16. Cppreference. Cppreference - complete online reference for the C and C++ languages and standard libraries. URL: <https://en.cppreference.com/w/>

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

[illegible]

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

Изм	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата