

**Московский авиационный институт
(национальный исследовательский университет)**

**Институт №8 «Компьютерные науки и прикладная
математика»**

**Кафедра 806 «Вычислительная математика и
программирование»**

Лабораторная работа №1 по курсу «Компьютерная графика»

Студент: Г. А. Ермеков

Преподаватель: В. Д. Бахарев

Группа: М8О-301Б-23

Дата:

Оценка:

Подпись:

Москва, 2025

Условие

Задача: В этой лабораторной работе вы познакомитесь с основами 3D-графики: построением простых 3D-объектов, проекцией на 2D-плоскость, а также научитесь работать с матрицами перспективы, ортографической проекции и аффинными преобразованиями.

1 Метод решения

Для реализации поставленной задачи была разработана программа на языке C++ с использованием графического API Vulkan.

Основные математические операции были реализованы вручную в файле `testbed/main.cpp`:

- Матрица проекции (*projection*) — формирует матрицу перспективной проекции на основе угла обзора (FOV), соотношения сторон и плоскостей отсечения.
- Аффинные преобразования — реализованы функции создания матриц трансляции (*translation*) и вращения (*rotation*) вокруг произвольной оси.
- Операции с матрицами — функция перемножения матриц (*multiply*) для комбинирования модельных преобразований.

Инициализация и работа с Vulkan (файл `source/veekay.cpp` и `testbed/main.cpp`):

- Создание `VkInstance`, `VkDevice`, `VkSwapchainKHR` для взаимодействия с оконной системой (GLFW).
- Загрузка шейдеров (`VkShaderModule`) и создание графического конвейера (`VkPipeline`). Вершинный шейдер (`shader.vert`) принимает матрицы трансформации и проекции через Push Constants.
- Управление памятью и буферами (`VkBuffer`, `VkDeviceMemory`) для хранения вершинных и индексных данных 3D-модели (куба).
- Синхронизация кадров с использованием `VkSemaphore` и `VkFence`.

В качестве интерактивного интерфейса для изменения параметров трансформации в реальном времени была интегрирована библиотека ImGui.

2 Результаты

В ходе выполнения лабораторной работы были изучены основы 3D-графики и принципы построения изображений с использованием графического API Vulkan.

Были реализованы и применены на практике:

- Математический аппарат для работы с матрицами 4x4 (проекция, вращение, перемещение).
- Базовая настройка графического конвейера Vulkan.

- Передача данных (вершин, матриц) в шейдеры.

Результатом работы является программа, отображающая 3D-куб с возможностью управления его вращением и положением.