Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Московский государственный технический университет имени Н.Э. Баумана»

(МГТУ им. Н.Э. Баумана)

Факультет: Информатика и системы управления

Кафедра: Теоретическая информатика и компьютерные технологии

**Лабораторная работа №7**

**«Оптимизация приложений OpenGL»**

**по курсу: «Компьютерная графика»**

Выполнил:

Студент группы ИУ9-42Б

Гасаев Г. К.

**Цель работы:**

* Изучить эффективные приемы организации приложений и оптимизации вызовов OpenGL;
* Оптимизировать приложение OpenGL, созданное в рамках лабораторной работы номер 6, на основе выбора наиболее эффективных методик;

* Использовать дисплейные списки, массивы вершин и еще 2 любые различные оптимизации (минимум 4 оптимизации);
* Оценить применимость выбранного метода оптимизации приложения OpenGL на основании измерения производительности;
* Результаты замеров оформить в табличном виде;

**Основная теория:**

**Процесс рисования призмы был переработан: вместо устаревших процедур glBegin, glEnd и glVertex использованы массивы вершин, индексов, а также нормалей и текстурных координат. Первым делом эти массивы рассчитываются и загружаются в GPU через VBO, а в дальнейшем призма отрисовывается через обращение к ним процедуры glDrawElements, получающей массив индексов. Для улучшения производительности использован связанный примитив GL\_TRIANGLE\_FAN.**

**Для доступа к необходимым функциям создания и работы с окном использовалась библиотека GLFW версии 3. Для работы с VBO - библиотека GLEW. Чтение текстур из файлов осуществляется посредством библиотеки stb\_image версии 2.28. Для дальнейшей работы - функции glGenTextures, glBindTexture и другие.**

**Расчёт скорости и положения свободно падающего на верхнее основание призмы источника света инкапсулирован в отдельном классе Point. Для удобства реализован класс трехмерного вектора и операции для него. Точечный локальный источник света приближается и отдаляется от верхнего основания призмы, падая и абсолютно упруго отскакивая. Процедурами OpenGL glLight\* были настроены фоновый свет и параметры освещения одного источника.**

**Команды отрисовки точки, представляющей свободно падающее тело, были собраны в дисплейный список. В целях оптимизации отключена нормализация векторов нормалей (GL\_NORMALIZE) и включено отсечение задних граней до передачи на этап растеризации (GL\_CULL\_FACE), отключено использование альфа-канала текстур (GL\_BLEND).**

**Для замеров времени работы использована библиотека time и создан класс TimeCounter, подсчитывающий среднее время обработки одного кадра сцены путем замера этого времени для установленного количества кадров и вычисления среднего арифметического.**

**Источники:**

[**https://www.khronos.org/opengl/**](https://www.khronos.org/opengl/)

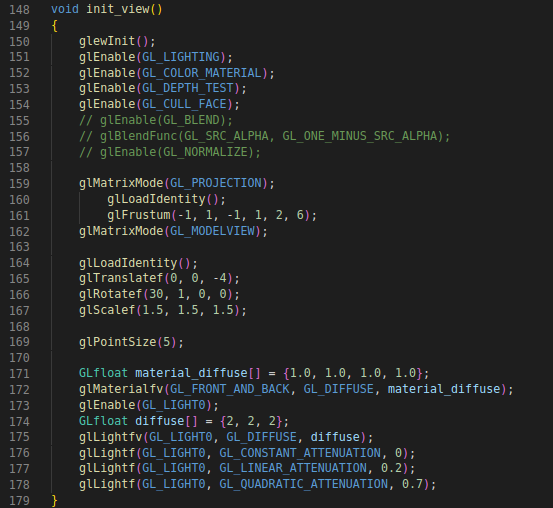
**http://aco.ifmo.ru/el\_books/computer\_visualization/lectures/10.html**

**https://habr.com/ru/articles/333932/**

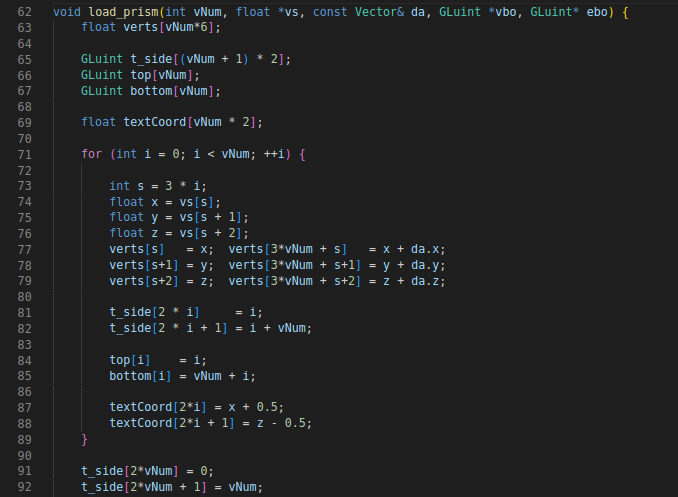
[**https://compgraphics.info/OpenGL/lighting/light\_sources.php**](https://compgraphics.info/OpenGL/lighting/light_sources.php)

**Практическая реализация:**

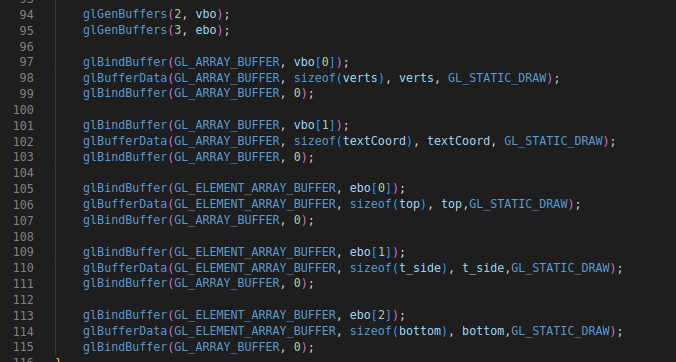
* **Настройки проекции, модельных преобразований и света:**



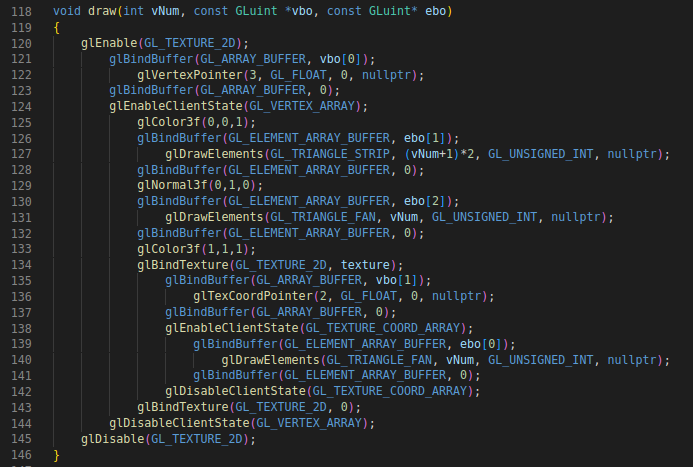
* **Создание массивов вершин, нормалей и текстурных координат призмы:**



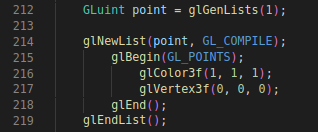
* **Создание VBO и загрузка в них соответствующих массивов:**



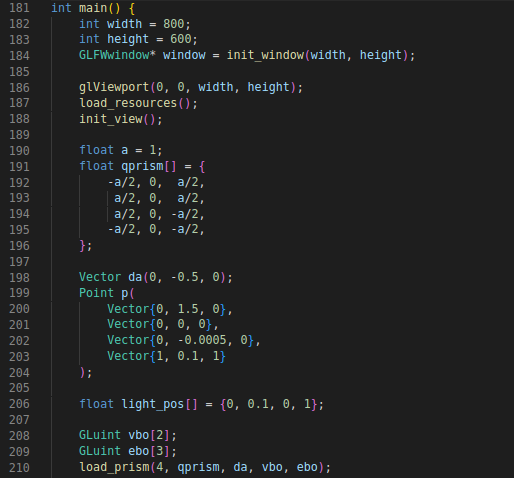
* **Отрисовка призмы:**

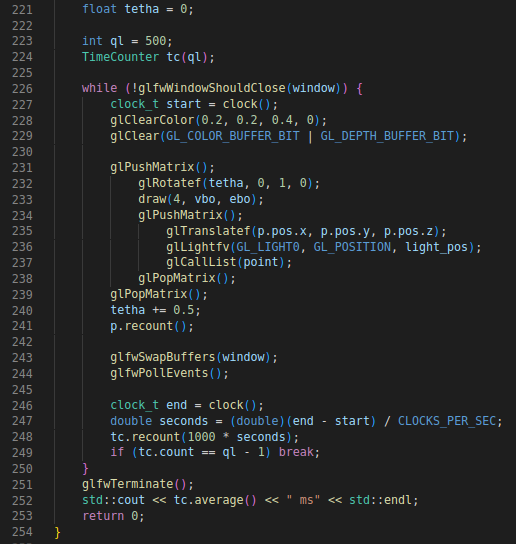


* **Создание дисплейного списка для отрисовки падающей точки:**

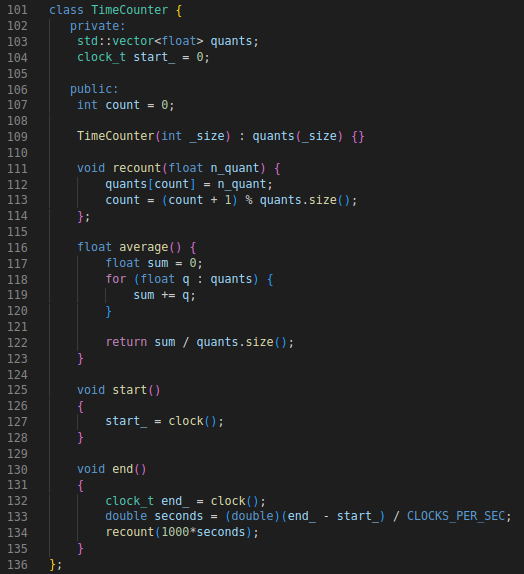


* **Главный цикл:**





* **Класс TimeCounter:**



**Заключение:**

**В ходе лабораторной работы:**

1. **Изучены различные методы оптимизации приложения.**
2. **Лабораторная работа номер 6 - рисование наклонная призма и использование освещения - была оптимизирована посредством:**

* **вершинных массивов;**
* **дисплейного списка;**
* **использования vbo;**
* **связанного примитива GL\_TRIANGLE\_FAN;**
* **отключения нормализации векторов;**
* **отсечения обратных граней до растеризации**

1. **Получена оценка производительности оптимизированной версии приложения на основе измерения времени работы и сравнения с таковым неоптимизированной версии:**

|  |  |  |
| --- | --- | --- |
| **Количество тиков\ Время работы (мс)** | **Обычная версия** | **Оптимизация** |
| **500** | **0.520552** | **0.504186** |
| **1000** | **0.510910** | **0.487836** |
| **1500** | **0.515781** | **0.491115** |

**Таким образом, увеличение эффективности можно проследить, даже несмотря на небольшие затраты времени на кадр в принципе и всевозможные погрешности измерений.**