МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВО «РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ГЕОЛОГОРАЗВЕДОЧНЫЙ УНИВЕРСИТЕТ ИМЕНИ СЕРГО ОРДЖИНЕКИДЗЕ»

Факультет: **Геологии и геофизики нефти и газа**

Кафедра: **Математики**

Курсовая работа по компьютерной графике на тему:

**«Модель молекулы»**

Студента группы ПМ-22: **Зуйкова Андрея Дмитриевича**

Руководитель: **Морочко А.Ф.**

**Дата: 19.12.2023**

**Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Морочко А.Ф./**

Москва 2023

**Оглавление**

1. Введение

1.1 Определение компьютерной графики и ее роль в современных приложениях.

1.2 Знакомство с OpenGL и его применением в различных областях.

1.3 Цели и задачи курсовой работы.

2. Обзор среды разработки и основных инструментов

2.1 Выбор и установка среды разработки OpenGL.

2.2 Основные инструменты и библиотеки для разработки с использованием OpenGL.

3. Заключение

3.1 Подведение итогов работы.

3.2 Выводы о полученных знаниях и навыках в области компьютерной графики.

4. Список литературы

**Введение**

**1.1 Компьютерная графика** — это область информационных технологий, которая занимается созданием, обработкой и визуализацией графических изображений на компьютере. Она объединяет методы и алгоритмы для создания и манипулирования графическими объектами, а также для их отображения на экране.

Роль компьютерной графики в современных приложениях очень важна и разнообразна. Некоторые из ее основных ролей включают:

1. Визуализация данных: Компьютерная графика позволяет представлять сложные данные в виде графиков, диаграмм, графов и других визуальных представлений. Это помогает пользователям лучше понимать и анализировать информацию.

2. Разработка компьютерных игр: Графика является одной из ключевых составляющих компьютерных игр. Она отвечает за создание виртуальных миров, персонажей, эффектов и анимации, что делает игры более реалистичными и привлекательными для игроков.

3. Компьютерное моделирование и визуализация: с помощью компьютерной графики можно создавать трехмерные модели объектов и сцен, проводить виртуальные эксперименты и анализировать результаты. Это находит применение в различных областях, таких как архитектура, инженерия, медицина, научные исследования и дизайн.

4. Компьютерная анимация и спецэффекты: Графика используется для создания анимации и спецэффектов в киноиндустрии, рекламе и мультимедиа. Она позволяет оживить статические изображения, создавать реалистические движения, взаимодействие объектов и визуальные эффекты, которые зачастую сложно или невозможно реализовать в реальной жизни.

5. Виртуальная реальность и дополненная реальность: Компьютерная графика играет ключевую роль в создании виртуальных и дополненных сред, где пользователи могут взаимодействовать с виртуальными объектами и сценами в реальном времени. Это находит применение в развлекательных приложениях, обучении, тренировках и симуляциях.

Это лишь некоторые из множества ролей компьютерной графики в современных приложениях. Она широко применяется в различных областях и продолжает развиваться, создавая новые возможности для визуализации и взаимодействия с информацией.

**1.2 OpenGL (Open Graphics Library)** — это кроссплатформенная графическая библиотека, которая обеспечивает программный интерфейс для создания 2D и 3D графики на компьютере. Она была разработана в 1992 году компанией Silicon Graphics и с тех пор стала одним из стандартов в индустрии компьютерной графики.

OpenGL используется в различных областях, включая:

1. Компьютерные игры: OpenGL является одним из основных API для создания игровой графики. Он позволяет разработчикам создавать реалистичные 3D миры, персонажей, эффекты и анимацию.

2. Виртуальная реальность: OpenGL используется для создания виртуальных сред, где пользователи могут взаимодействовать с виртуальными объектами и сценами в реальном времени.

3. Научная визуализация: OpenGL широко используется в научных и инженерных приложениях для визуализации сложных данных и моделей.

4. Архитектура и дизайн: OpenGL используется для создания визуальных прототипов, планирования и проектирования зданий, мебели и других объектов.

5. Мультимедиа и анимация: OpenGL используется для создания анимации, спецэффектов и другой графики в киноиндустрии, рекламе и мультимедиа.

6. Обучение и тренировки: OpenGL может использоваться в различных приложениях для обучения и тренировок, таких как симуляторы полетов и вождения, медицинские симуляторы и игры-тренажеры.

OpenGL поддерживает широкий диапазон функций, включая работу с текстурами, освещением, материалами, прозрачностью, тенями, перспективой и многими другими. Он также позволяет программистам использовать GPU для ускорения процесса отображения графики.

В целом, OpenGL является мощным инструментом для создания 2D и 3D графики на компьютере и находит широкое применение в различных областях.

**1.3 Цель исследования:**

Данная курсовая работа направлена на изучение принципов работы с библиотеками OpenGL и PyGame с целью создания модели молекулы воды. Основной задачей является приобретение практических навыков работы с трехмерной графикой, текстурами и интерактивностью при помощи указанных инструментов.

Задачи исследования:

1. Изучение основных принципов рендеринга 3D-графики с использованием библиотеки OpenGL.

2. Создание трехмерной модели молекулы воды с применением текстур для достижения реалистичного визуального эффекта.

3. Разработка интерактивной среды при помощи библиотеки PyGame для возможности взаимодействия с моделью молекулы (например, вращение и масштабирование).

4. Повышение навыков программирования на языке Python в контексте работы с графическими библиотеками.

Ожидаемые результаты:

В результате выполнения курсовой работы ожидается получение практических навыков работы с трехмерной графикой, текстурами и интерактивностью при разработке визуальных проектов. Полученные знания и опыт могут быть применены в области разработки игр, симуляций и других визуальных приложений.

**Обзор среды разработки и основных инструментов**

2.1 В качестве среды разработки был выбран VSCode, из-за его многочисленных плюсов:

1. Легковесность и производительность: VSCode разработан таким образом, чтобы быть легким и быстрым. Он имеет минимальное использование ресурсов и быстро запускается, что позволяет сосредоточиться на разработке без задержек.

2. Многоязыковая поддержка: VSCode поддерживает широкий спектр языков программирования, включая JavaScript, Python, C++, Java, HTML, CSS и многие другие. Он предлагает автодополнение кода, подсветку синтаксиса и функции проверки ошибок для каждого языка.

3. Расширяемость: Одна из самых привлекательных особенностей VSCode - его расширяемость. С помощью множества доступных расширений и плагинов от сообщества разработчиков, можно настроить среду разработки под свои потребности. Это позволяет добавлять новые функции, интегрироваться с другими инструментами и создавать собственные расширения.

4. Интеграция системы контроля версий: VSCode имеет встроенную поддержку систем контроля версий, таких как Git. Это позволяет разработчикам работать с репозиториями, просматривать изменения, фиксировать коммиты и решать конфликты слияния, все изнутри среды разработки.

5. Отладка и профилирование: VSCode предлагает инструменты отладки, которые помогают разработчикам находить и исправлять ошибки в коде. Он поддерживает различные языки программирования и предоставляет возможность устанавливать точки останова, выполнять пошаговое выполнение и анализировать значения переменных.

6. Интеграция с облаком: VSCode интегрируется с различными облачными сервисами, такими как Azure и AWS. Это обеспечивает доступ к облачным ресурсам, развертывание и управление приложениями, а также интеграцию с другими сервисами для более удобной разработки и развертывания проектов.

7. Активное сообщество: Одной из причин популярности VSCode является его активное сообщество пользователей и разработчиков. Есть множество ресурсов, форумов, расширений и тем, доступных для обмена опытом и получения поддержки при необходимости.

В целом, VSCode - это мощная и гибкая среда разработки, которая обеспечивает эффективное программирование на различных языках, интеграцию с другими инструментами и расширяемость посредством плагинов. Она позволяет разработчикам ускорить и улучшить свой рабочий процесс, делая его более продуктивным.

2.2 Для разработки были использованы, некоторые модули, такие как:

PyOpenGL - это привязка к OpenGL для языка Python. Она предоставляет доступ к основным функциям OpenGL и дополнительным расширениям. PyOpenGL позволяет вам создавать и управлять контекстами OpenGL, загружать текстуры, работать с шейдерами и многое другое.

GLFW - как и в случае с разработкой на других языках, библиотека GLFW может быть использована для создания окон и обработки пользовательского ввода. Существует версия GLFW для Python (PyGLFW), которая обеспечивает простой доступ к функциям GLFW из языка Python.

Pygame - это библиотека для разработки 2D-игр на Python, но она также может быть использована для работы с OpenGL. Pygame предоставляет функции для создания окон, обработки событий, загрузки и отображения изображений, а также для работы с звуком. Она также имеет модуль OpenGL, который позволяет использовать функциональность OpenGL вместе с Pygame.

**Заключение**

3.1 В итоге данной курсовой работы мы рассмотрели основы разработки с использованием OpenGL на языке Python. Мы изучили различные библиотеки, которые могут быть использованы для работы с OpenGL, такие как PyOpenGL, Pygame и GLFW.

Мы обсудили основные концепции и функции OpenGL, такие как вершины, индексы, шейдеры и матрицы. Также были представлены примеры кода, демонстрирующие создание окна, загрузку и отображение текстур, выполнение преобразований и трансформаций, а также рендеринг 2D и 3D графики.

3.2 В целом, данная работа предоставляет основы для начала разработки приложений с использованием OpenGL на языке Python. Однако, для более глубокого понимания и продвинутого использования OpenGL, рекомендуется дополнительное изучение документации и руководств, а также практическое применение полученных знаний.

**Список литературы**

1. <https://pygame-docs.website.yandexcloud.net/ref/pygame.html>
2. <https://pygame.readthedocs.io/_/downloads/en/latest/pdf/>
3. <https://pyopengl.sourceforge.net/documentation/>
4. <https://habr.com/ru/articles/246625/>
5. <https://github.com/GoldenMan123/pyopengl/blob/master/documentation.pdf>