**Звіт**-**ретроспектива до лабораторної №3 з ОООП**

**Адамов О.В. К-28**

Я використав OpenGL(та GLSL – мова для програмування шейдерів OpenGL), GLFW, GLEW, GLM.

Посилання на бібліотеки:

GLFW - <https://www.glfw.org/>

GLEW - <http://glew.sourceforge.net/>

GLM - <https://github.com/g-truc/glm>

Документація OpenGL, яку я використовував - <https://docs.gl/>

Документація GLSL, яку я використовував - <https://docs.gl/sl4/all>

Відповіді на запитання:

1. Які конкретні задачі планували вирішувати за допомогою цієї бібліотеки?

Я хотів візуалізувати результати алгоритмів з проекту на кожному кроці, для цього вирішив використати OpenGL.

GLFW я використав для того щоб створити вікно (воно створюється по різному, на різних ОС, я хотів щоб візуалізація була кросплатформною, але не хотів писати купу коду для різних ОС).

GLEW я використав, щоб отримати можливість користуватися сучасним OpenGl. Ця бібліотека декларує функції OpenGL та отримує їх визначення з драйверів відеокарти користувача.

GLM я використав для того, щоб візуалізація була стійкою до змінення розмірів вікна, за допомогою ортогонального проектування.

1. Чому було обрано саме цю бібліотеку, а не аналоги?

Я вирішив використати не бібліотеку з готовою візуалізацію, а щось більш нізкорівневе, щоб розібратися з основами графічного програмування, та написати візуалізацію самому на основі отриманих знань. Було обрано саме OpenGL, а не DirectX, чи Vulkan, чи ще щось схоже, через те що OpenGL дещо простіше у використанні, по ньому досить багато навчального матеріалу, та він кросплатформний.

GLFW та GLEW популярніші та простіші у використанні ніж аналоги.

GLM це математична бібліотека, яка “заточена” під OpenGL. Не потрібно виконувати зайві дії, наприклад транспонування матриць, щоб OpenGL(GLSL) правильно з ними працювали. Так як я планував використовувати тільки OpenGL для візуалізації GLM – вдалий вибір.

1. Наскільки просто та зрозуміло було отримати, встановити, налаштувати та почати використовувати цю бібліотеку?

Метою було використувувати OpenGL. І добитися цього не так вже і легко, потрібно встановити GLEW, і ініціалізувати його уже в контексті OpenGL, для цього потрібно було встановити GLFW, ініціалізувати її, створити вікно та контекст. Благо приклади ініціалізації були у документації бібліотек, тому це було не складно. З GLM не виникло ніяких труднощів.

1. Наскільки зрозумілою та корисною була документація бібліотеки?

Документація GLFW, GLEW, та GLFW – була цілком зрозумілою.

Документація OpenGL в цілому добра, але пошук ведеться тільки за функціями, і було проблематично знайти що позначають ти чи інші константи. Знайти константи OpenGL я зміг у wikipedia.

Документація GLSL взагалі не містила потрібної мені інформації. Там був перелік функцій, а я хотів знайти які шейдери, що приймають, та що виводять. Знайти цю інформацію я зміг у wikipedia.

1. Наскільки було зрозуміло, як саме використовувати бібліотеку, які класи/методи/функції використовувати для вирішення поставлених задач?

Без туторіалів було б важко розібратися. Тільки щоб нарисувати один трикутник потрібно годину-дві розбиратись та все сетапити.

1. Наскільки зручно було використовувати бібліотеку, чи не треба було писати багато надлишкового коду?

Потрібно було писати досить багато надлишкового коду щоб запустити свій шейдер, та щоб отримати інформацію про помилки, все інше більш-менш зручно.

1. Наскільки зрозумілою була поведінка класів/методів/функцій з бібліотеки?

Якщо розібратися з специфікую роботи OpenGL – state machine, то поведінка більшості функцій є зрозумілою.

1. Наскільки зрозумілою була взаємодія між різними класами/методами/функціями цієї бібліотеки, а також взаємодія між бібліотекою та власним кодом?

Якщо розібратися з специфікую роботи OpenGL – state machine, то все зрозуміло.

1. Чи виникали якісь проблеми з використанням бібліотеки? Чи вдалось їх вирішити, як саме?

Серйозних проблем не виникло.

1. Що хорошого можна сказати про цю бібліотеку, які були позитивні аспекти використання бібліотеки?

Завдяки використанню цих бібліотек, я почав розуміти основи графічного програмування. Також я зміг за допомогою цих засобів досягти мети, що є головним фактором, у оцінці бібліотек.

1. Що поганого можна сказати про цю бібліотеку, які були негативні аспекти використання бібліотеки?

Те що потрібно писати багато коду для того щоб запустити шейдер я ще можу зрозуміти. Але чому потрібно писати свою функцію з якимись страшним ім’ям та параметрами, або незручні дефайни, щоб отримати інформацію про помилки у функціях OpenGL я не розумію. Це можна віднести до мінусів OpenGL.

1. Якби довелось вирішувати аналогічну задачу, але вже враховуючи досвід використання в цій лабораторній роботі, що варто було б робити так само, а що змінити? Можливо, використати інші бібліотеки, чи використати інші можливості цієї бібліотеки, чи інакше організувати код, чи ще щось?

Я б використовував ті ж самі засоби. Але якщо потрібно було б писати більше шейдерів, більше Vertex buffers, то є сенс їх більш абстрагувати.

Окрім документації головним джерелом інформації був цей плейліст:

<https://www.youtube.com/watch?v=W3gAzLwfIP0&list=PLlrATfBNZ98foTJPJ_Ev03o2oq3-GGOS2>