

Лабораторная работа №1

Тема: Построение изображений 2D-кривых.

Задача: Написать и отладить программу, строящую изображение заданной замечательной кривой.

Вариант №22: $\rho = a \cdot \sin(6\phi)$

1 Решение

Для реализации задачи по построению изображения заданной замечательной кривой в полярных координатах были использованы модули «numpy» и «matplotlib». Модуль «numpy» предоставляет инструменты для численных вычислений, а «matplotlib» - для визуализации данных.

Центральной частью решения является функция, которая вычисляет значения ρ для заданной функции $\rho = a \times \sin(6\phi)$ на основе массива значений ϕ , который генерируется с помощью «numpy».

Основная сложность заключается в создании интерактивного графика, который позволит пользователю динамически изменять значение параметра a . Это достигается с помощью виджета «Slider» из «matplotlib.widgets».

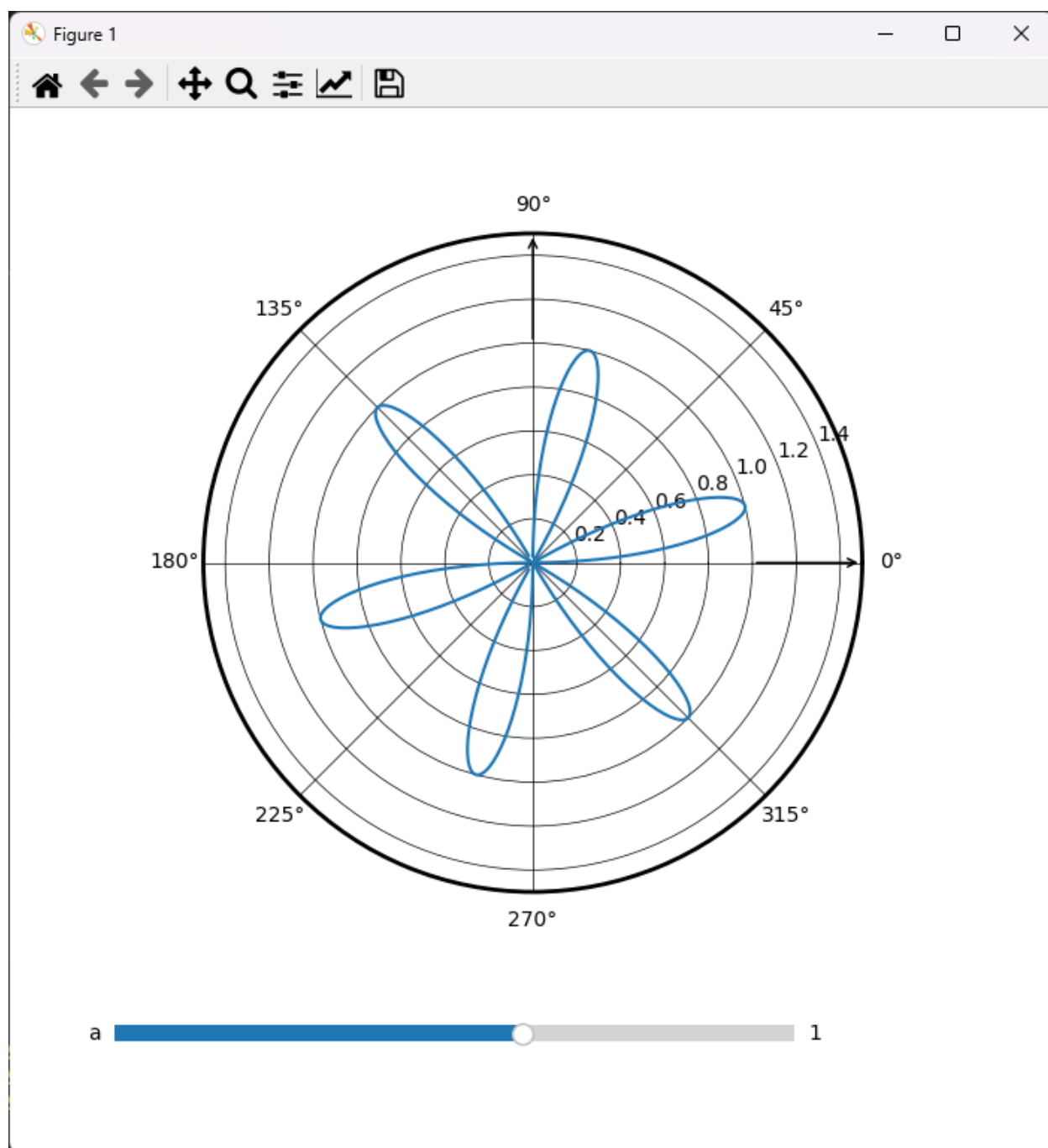
После вычисления начальных значений ρ и ϕ , кривая отображается на полярном графике. Затем добавляется слайдер, который позволяет изменять значение a в реальном времени, что приводит к пересчету и перерисовке кривой.

Для улучшения визуального восприятия были внесены следующие изменения:

1. Яркие выраженные оси с четкими делениями.
2. Стрелки, указывающие направление осей.

3. Динамические пределы для радиальной оси, чтобы кривая всегда отображалась целиком.

Таким образом, с помощью сочетания численных вычислений и визуализации была создана интерактивная программа, позволяющая пользователям наблюдать и анализировать поведение кривой при изменении параметра a .



2 Вывод

В ходе реализации программы для построения замечательной кривой в полярных координатах были успешно использованы модули `numpy` и `matplotlib` для численных вычислений и визуализации графиков соответственно. Проект подчеркивает важность инкапсуляции и модульности, поскольку основные функции, такие как вычисление кривой и ее отображение, были четко разделены.

Особое внимание было уделено интерактивности графика, что позволило пользователю в реальном времени изменять параметры кривой и наблюдать за ее поведением. Добавление элементов, таких как ярко выраженные оси, стрелки направления и динамические пределы, значительно улучшило визуальное восприятие и понимание кривой

Этот проект демонстрирует, как комбинация правильных инструментов и внимание к деталям могут привести к созданию эффективной и информативной визуализации данных.

Лабораторная работа №2

Тема: Каркасная визуализация выпуклого многогранника. Удаление невидимых линий.

Задача: Разработать формат представления многогранника и процедуру его каркасной отрисовки в ортографической и изометрической проекциях. Обеспечить удаление невидимых линий и возможность пространственных поворотов и масштабирования многогранника. Обеспечить автоматическое центрирование и изменение размеров изображения при изменении размеров окна.

Вариант №22: правильный додекаэдр.

1 Решение

В данной программе реализована визуализация додекаэдра и его проекций с использованием модулей `numpy` и `matplotlib`. `numpy` применяется для численных вычислений, включая определение координат вершин додекаэдра и матриц вращения, тогда как `matplotlib` используется для создания 3D и 2D визуализаций.

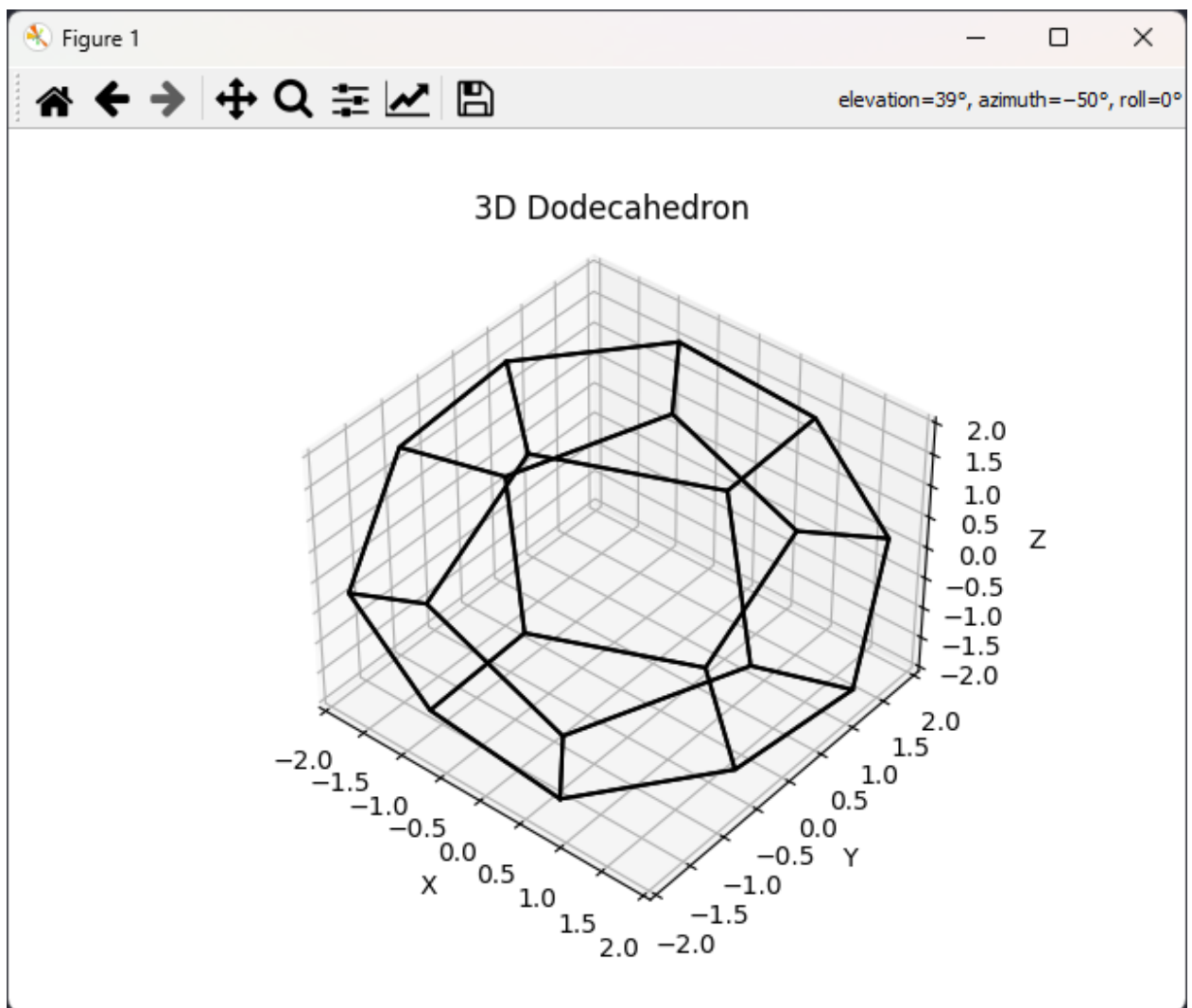
Центральной частью программы являются функции, отвечающие за визуализацию додекаэдра в трехмерном пространстве и его проекции в двухмерных плоскостях. Программа начинается с определения координат вершин додекаэдра и золотого сечения, которые затем используются для построения 3D модели. Для реализации вращения додекаэдра создана функция `rotation_matrix`, а `scale_vertices` применяется для масштабирования вершин.

Особенностью программы является визуализация додекаэдра с учетом удаления невидимых линий, что достигается путем сортировки граней по средней глубине. Также реализованы функции для создания ортографических и изометрических проекций додекаэдра.

Одной из ключевых особенностей является интерактивность, позволяющая пользователю изменять масштаб 3D модели додекаэдра с помощью клавиш. Это достигается через функции, которые реагируют на ввод пользователя и перерисовывают фигуру в новом масштабе.

Для повышения удобства восприятия и улучшения визуализации в программе учтены такие аспекты, как четко обозначенные оси, равномерное масштабирование и учет перспективы при проекции.

Таким образом, данная программа представляет собой сложное сочетание численных вычислений и визуализации, позволяя наблюдать и анализировать трехмерную структуру додекаэдра и его двухмерные проекции.



2 Вывод

В ходе создания программы для визуализации додекаэдра и его проекций были эффективно применены модули `numpy` и `matplotlib`. `numpy` использовался для точных численных расчетов, включая определение вершин додекаэдра и матриц вращения, в то время как `matplotlib` обеспечил мощные средства для создания как трехмерных, так и двухмерных визуализаций.

Проект выделяется своей модульностью и четким разделением функций, что облегчает понимание и расширение кода. Включение таких элементов, как вращение додекаэдра, масштабирование, а также ортографическая и изометрическая проекции, показывает глубину и многообразие применения геометрических и визуализационных методов.

Особое внимание было уделено интерактивности программы, позволяющей пользователю в реальном времени взаимодействовать с трехмерной моделью, что значительно усиливает образовательный и практический потенциал приложения. Наглядное представление трехмерных объектов и их проекций способствует лучшему пониманию сложных геометрических структур.

В целом, этот проект является отличным примером того, как сочетание продуманных алгоритмов, инструментов визуализации и интерактивности может привести к созданию интуитивно понятной и полезной образовательной программы.

Лабораторная работа №3

Тема: Основы построения фотореалистичных изображений.

Задача: Используя результаты Л.Р. 2, аппроксимировать заданное тело выпуклым многогранником. Точность аппроксимации задается пользователем. Обеспечить возможность вращения и масштабирования многогранника и удаление невидимых линий и поверхностей. Реализовать простую модель закраски для случая одного источника света. Параметры освещения и отражающие свойства материала задаются пользователем в диалоговом режиме.

Вариант №22: бочка.

1 Решение

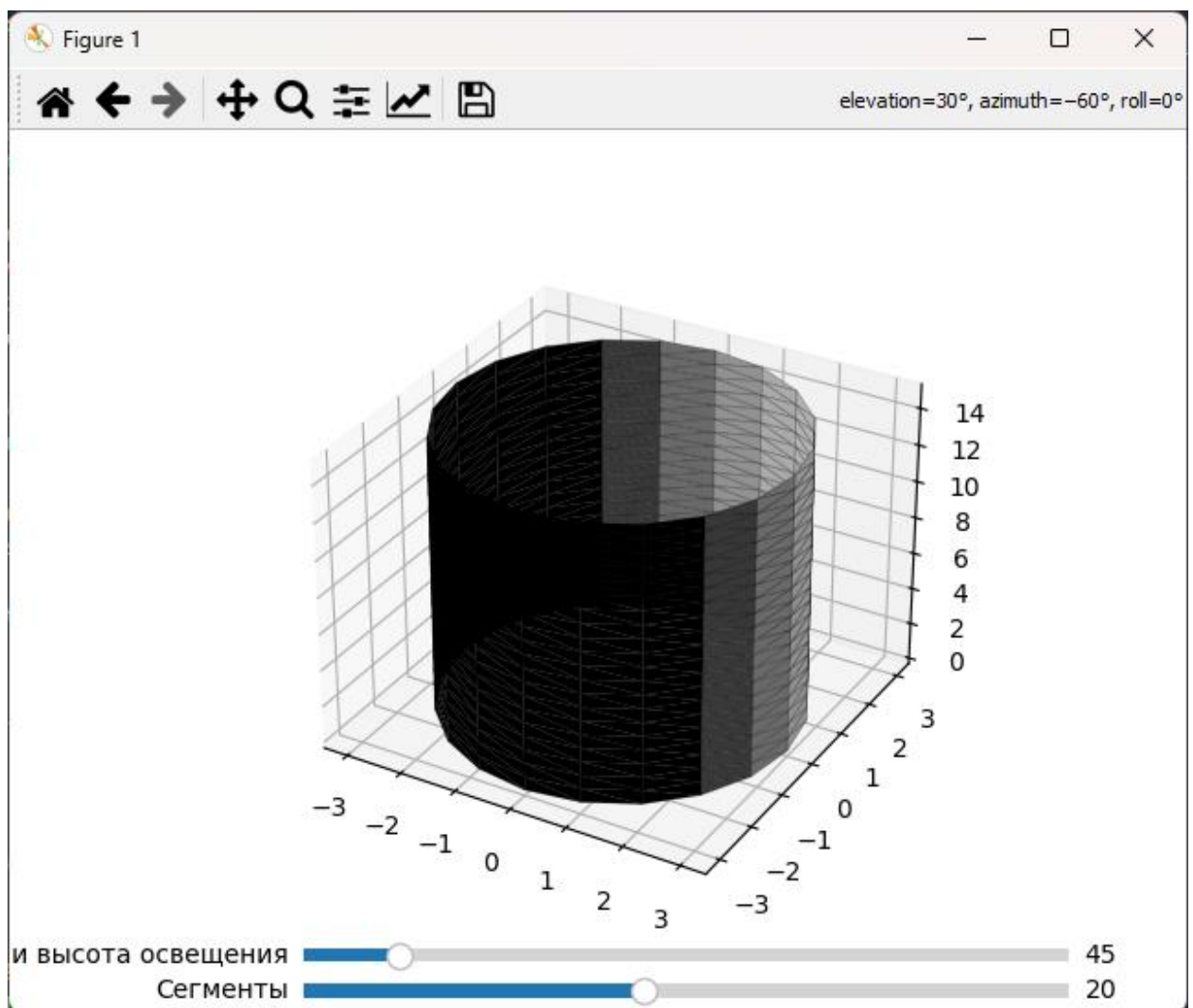
В данной программе реализована задача построения и визуализации 3D модели цилиндрического барреля с использованием модулей `numpy` и `matplotlib`. `numpy` используется для численных вычислений, включая генерацию вершин и граней барреля, а `matplotlib` - для их визуализации и создания интерфейса взаимодействия с пользователем.

Центральными элементами программы являются функции для создания вершин и граней цилиндра, а также для их отрисовки с учетом освещения. Особое внимание уделено созданию реалистичного эффекта освещения с использованием класса `LightSource` из `matplotlib`. Освещение изменяется динамически в зависимости от положения источника света, что задается пользователем через интерфейс.

Интерактивность обеспечивается с помощью виджетов `Slider`, которые позволяют пользователю изменять количество сегментов барреля и положение источника света в реальном времени. Это приводит к динамическому пересчету вершин и граней и их последующей перерисовке.

В программе также учтены аспекты, улучшающие визуальное восприятие и понимание модели, включая четко обозначенные оси и настройку масштаба визуализации.

В результате, программа представляет собой комплексное сочетание численных вычислений, визуализации и интерактивных элементов управления, позволяющее наглядно наблюдать и анализировать поведение 3D модели барреля при изменении его параметров.



2 Вывод

В процессе разработки программы для визуализации 3D модели цилиндрического барреля были успешно задействованы модули `numpy` и `matplotlib`, обеспечивающие численные вычисления и визуализацию

соответственно. Этот проект подчеркивает важность структурированного подхода в программировании, так как основные функции, такие как генерация вершин и граней барреля, их отрисовка с учетом освещения и интерактивное управление, были четко разграничены.

Особое внимание уделено интерактивности, которая позволяет пользователю в реальном времени изменять количество сегментов барреля и положение источника света, что приводит к немедленному обновлению визуализации. Эта возможность значительно усиливает образовательный и практический потенциал программы, делая ее более наглядной и понятной для пользователя.

Проект демонстрирует, как сочетание правильных вычислительных инструментов и детализированного внимания к элементам визуализации и интерактивности может привести к созданию эффективной и информативной программы для визуализации сложных трехмерных объектов. Это подтверждает, что тщательно спланированный подход к разработке и внимание к пользовательскому опыту могут значительно улучшить конечный продукт.

Лабораторная работа №4-5

Тема: Ознакомление с технологией OpenGL.

Задача: Создать графическое приложение с использованием OpenGL. Используя результаты Л.Р. 3, изобразить заданное тело (то же, что и в л.р. 3) с использованием средств OpenGL 2.1. Использовать буфер вершин. Точность аппроксимации тела задается пользователем. Обеспечить возможность вращения и масштабирования многогранника и удаление невидимых линий и поверхностей. Реализовать простую модель освещения на GLSL. Параметры освещения и отражающие свойства материала задаются пользователем в диалоговом режиме.

Вариант №22: бочка.

1 Решение

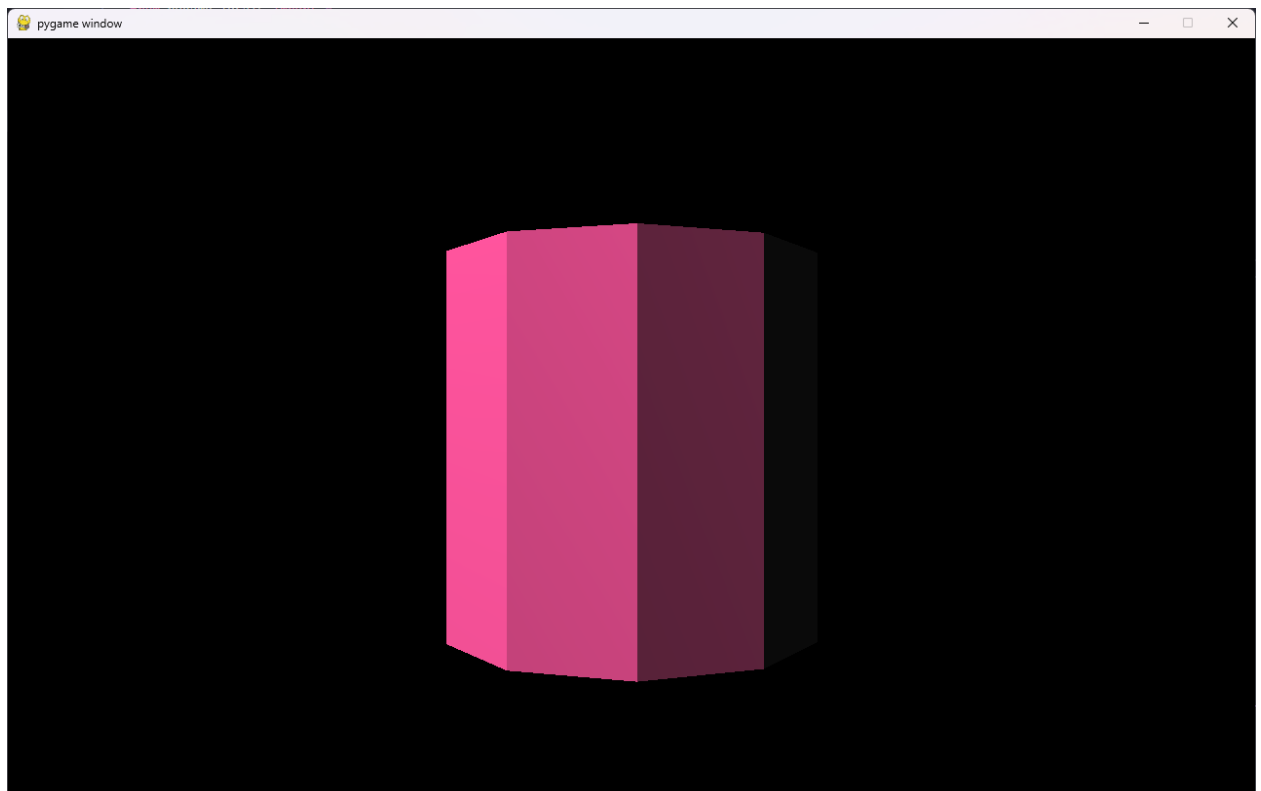
В данной программе реализована задача построения и визуализации 3D модели цилиндра с использованием модулей pygame, OpenGL и numpy. Модуль numpy используется для численных вычислений, связанных с генерацией вершин и граней цилиндра. pygame и OpenGL обеспечивают визуализацию и интерактивность модели в трехмерном пространстве.

Центральными элементами программы являются функции для создания вершин и граней цилиндра, а также для их отрисовки с учетом освещения и нормалей. Особое внимание уделено взаимодействию с пользователем: в программе реализованы функции для вращения модели с помощью мыши и изменения параметров модели, таких как точность отображения и уровень отражения света, через клавиатурные команды.

Сложность проекта заключается в создании интерактивной 3D сцены, которая позволяет пользователю манипулировать объектом и наблюдать за изменениями в реальном времени. Основные изменения включают в себя

динамическое управление количеством сегментов цилиндра и настройку параметров освещения.

Проект демонстрирует, как сочетание численных вычислений, графического рендеринга и интерактивного взаимодействия может создать образовательную и занимательную среду для изучения трехмерных объектов и основ освещения в компьютерной графике.



2 Вывод

В ходе разработки программы для визуализации 3D модели цилиндра были успешно применены модули `numpy`, `pygame` и `OpenGL`. `numpy` использовался для точных численных расчетов, связанных с созданием вершин и граней цилиндра, в то время как `pygame` и `OpenGL` обеспечили визуализацию и интерактивность в трехмерном пространстве.

Проект подчеркивает важность четкой структуризации и инкапсуляции функций, так как были разработаны отдельные функции для генерации вершин, граней, расчета нормалей и отрисовки 3D модели. Интерактивность

программы, позволяющая пользователю манипулировать объектом и изменять его параметры в реальном времени, играет ключевую роль в образовательном и практическом потенциале программы.

Элементы управления, такие как вращение модели мышью и настройка параметров через клавиатуру, а также реализация динамического освещения и отражения, значительно улучшили визуальное восприятие и понимание модели.

В итоге, проект демонстрирует, как сочетание численных вычислений, 3D визуализации и интерактивного взаимодействия может создать эффективную и информативную среду для изучения основ компьютерной графики и трехмерного моделирования.

Лабораторная работа №6

Тема: Создание шейдерных анимационных эффектов в OpenGL 2.1.

Задача: Для поверхности, созданной в л.р. 5, обеспечить выполнение следующего шейдерного эффекта.

Вариант №2: Анимация. Цветовые координаты изменяются по синусоидальному закону.

1 Решение

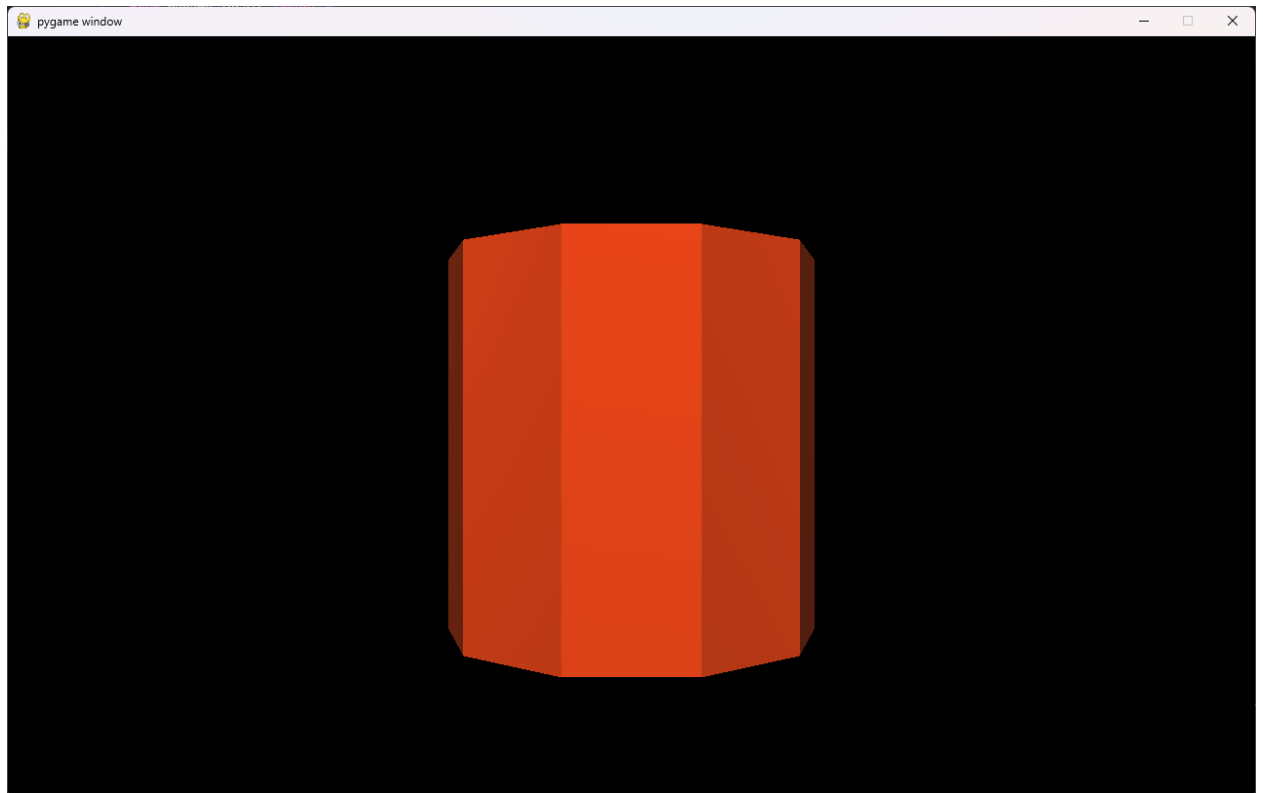
В данной программе реализована задача построения и визуализации 3D модели цилиндра с использованием модулей pygame, OpenGL и numpy. numpy предоставляет инструменты для численных вычислений, связанных с генерацией вершин и граней цилиндра. pygame и OpenGL используются для создания интерактивной 3D сцены и визуализации модели.

Основным элементом решения является функция для генерации вершин и граней цилиндра, а также функция для их отрисовки в 3D пространстве. Ключевой особенностью программы является интерактивность: пользователь может управлять вращением модели с помощью мыши и изменять параметры модели, такие как точность отображения и уровень отражения света, через клавиатурные команды.

Сложность проекта заключается в создании динамически изменяемой 3D сцены, которая позволяет пользователю взаимодействовать с объектом и наблюдать за изменениями в реальном времени. Особое внимание уделено освещению и материалам, чтобы улучшить реалистичность и визуальное восприятие модели.

Таким образом, программа представляет собой сложное сочетание численных вычислений, 3D визуализации и интерактивных элементов

управления, позволяющее наглядно наблюдать и анализировать особенности трехмерных объектов и основ освещения в компьютерной графике.



2 Вывод

В процессе разработки программы для визуализации 3D модели цилиндра были эффективно использованы модули `numpy`, `pygame` и `OpenGL`. `numpy` обеспечил необходимые численные вычисления для генерации вершин и граней цилиндра, в то время как `pygame` и `OpenGL` были задействованы для создания интерактивной трехмерной сцены и визуализации модели.

Проект подчеркивает важность четкой структуризации и модульного подхода, поскольку были разработаны отдельные функции для каждого аспекта моделирования и визуализации. Особое внимание уделено интерактивным элементам, таким как управление вращением модели с помощью мыши и изменение параметров модели через клавиатурные команды, что позволило пользователям наблюдать за изменениями в реальном времени.

Добавление динамических элементов, таких как изменение освещения и материала в зависимости от времени, значительно улучшило визуальное восприятие и понимание модели. Это подчеркивает, как правильное сочетание инструментов и внимание к деталям могут создать эффективную и информативную среду для исследования основ компьютерной графики и трехмерного моделирования.

Лабораторная работа №7

Тема: Построение плоских полиномиальных кривых.

Задача: Написать программу, строящую полиномиальную кривую по заданным точкам. Обеспечить возможность изменения позиции точек и, при необходимости, значений касательных векторов и натяжения.

Вариант №22: Сегмент кубического сплайна по конечным точкам и касательным.

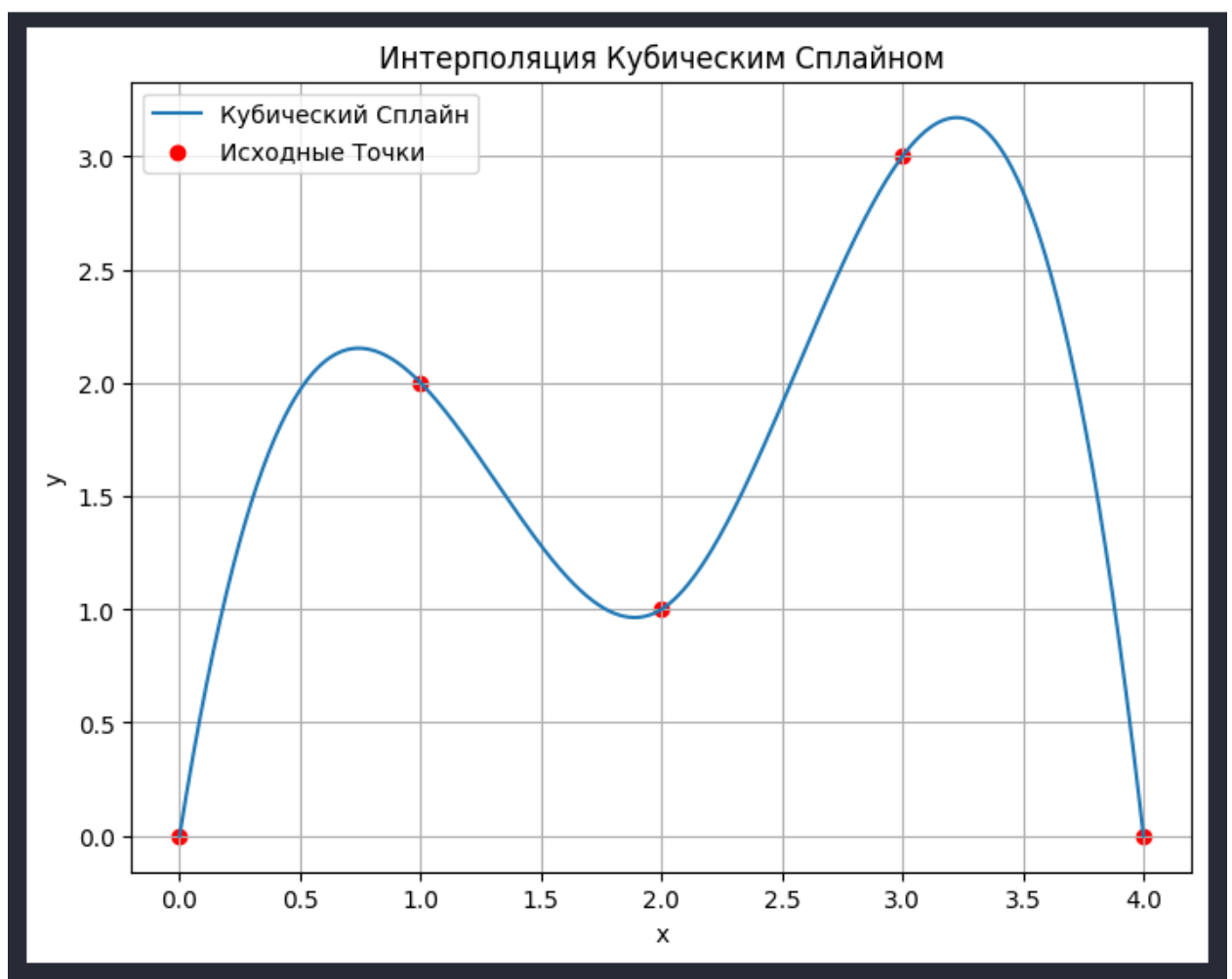
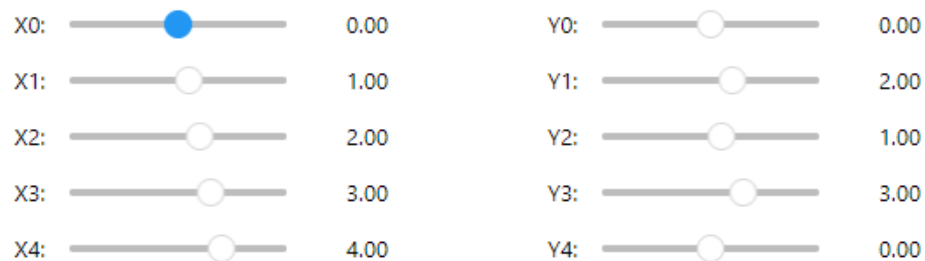
1 Решение

В этой программе реализована задача построения и визуализации кубического сплайна, используя модули `numpy`, `matplotlib` и `ipywidgets`. Модуль `numpy` применяется для численных вычислений, связанных с генерацией и обработкой точек данных, в то время как `matplotlib` используется для визуализации кубического сплайна. `ipywidgets` предоставляет интерактивные элементы управления для динамического изменения точек данных.

Центральными частями решения являются функции для генерации кубического сплайна на основе заданных точек и для его визуализации вместе с исходными точками. Основная сложность заключается в создании интерактивной среды, позволяющей пользователю изменять положение точек с помощью ползунков. Это достигается за счет использования виджетов `ipywidgets`, которые обновляют визуализацию сплайна в реальном времени при изменении положения точек.

После задания начальных точек и создания виджетов для их изменения, кривая сплайна отображается на графике. Пользователь может интерактивно регулировать положение каждой точки, что ведет к немедленному пересчету и перерисовке кривой сплайна.

Таким образом, программа является примером сочетания численных вычислений, визуализации и интерактивности, позволяя пользователю наглядно исследовать и анализировать свойства кубического сплайна при изменении ключевых точек.



2 Вывод

В процессе разработки программы для визуализации кубического сплайна были успешно задействованы модули `numpy`, `matplotlib` и `ipywidgets`.

numpy использовался для численных вычислений, связанных с созданием и обработкой точек данных, в то время как matplotlib обеспечил визуализацию сплайна. ipywidgets добавил в проект важный элемент интерактивности, позволяя пользователю динамически изменять точки и наблюдать за их влиянием на кривую сплайна в реальном времени.

Проект подчеркивает важность инкапсуляции и модульности в программировании: отдельные функции были разработаны для генерации сплайна и его отрисовки. Интерактивность, реализованная через виджеты, позволяла пользователям непосредственно взаимодействовать с моделью, изменяя положение точек и наблюдая за последствиями этих изменений на форму кривой.

Добавление деталей, таких как четко обозначенные оси, легенда и сетка, улучшило визуальное восприятие и понимание кривой. В целом, проект демонстрирует, как сочетание правильных вычислительных инструментов и внимание к деталям интерактивности могут создать мощное средство для визуализации и анализа сложных математических моделей.