**Документация разработчика**

**Текст программы**

**Наименование программы**

**3DObjectVisualizationApp** — приложение для визуализации трехмерного объекта (каркасная модель) на языке C++ с использованием фреймворка Qt 6.0.1 и модуля Qt Quick 3D.

**Область применения**

Программа предназначена для создания и визуализации трехмерных объектов. Она может быть использована в учебных целях для изучения методов и алгоритмов трехмерной визуализации.

**Назначение программы**

* Создание и визуализация трехмерного изображения первой буквы фамилии.
* Реализация трехмерных преобразований объекта (масштабирование, перенос, вращение вокруг произвольной оси).
* Реализация трех ортографических проекций заданного объекта (на координатные плоскости Oxy, Oxz, Oyz).

**Функциональные возможности**

* Ввод и отображение координат вершин трехмерного объекта.
* Выполнение трехмерных преобразований объекта.
* Отображение итоговой матрицы преобразования.
* Визуализация трех ортографических проекций объекта.

**Описание программы**

**Структура программы**

* **3DObjectVisualizationApp**: главный класс приложения, наследующийся от QMainWindow.
  + **Методы инициализации**:
    - \_\_init\_\_: инициализация приложения, установка заголовка, размеров окна и вызов методов для создания виджетов и настроек.
    - create\_widgets: создание элементов управления (кнопки, поля ввода и т.д.).
    - setup\_canvas: настройка холста для рисования и отображение координатной сетки.
    - draw\_grid: отрисовка системы координат и сетки на холсте.
  + **Методы обработки событий**:
    - transform\_object: выполнение трехмерных преобразований объекта и отображение результатов.
    - draw: отрисовка трехмерного объекта и его проекций.
  + **Алгоритмы преобразования**:
    - scale\_object: реализация масштабирования объекта.
    - translate\_object: реализация переноса объекта.
    - rotate\_object: реализация вращения объекта вокруг произвольной оси.
  + **Вспомогательные методы**:
    - read\_input: чтение входных данных (координат вершин) из файла.
    - draw\_vertex: отрисовка вершины на холсте.
    - draw\_edge: отрисовка ребра на холсте.

**Используемые библиотеки и модули**

* **Qt 6.0.1**: фреймворк для создания графического интерфейса пользователя.
  + **QMainWindow**: класс для создания главного окна приложения.
  + **QPainter**: класс для рисования графических элементов.
  + **QFile**: класс для работы с файлами.
  + **QFileDialog**: класс для отображения диалоговых окон для открытия файлов.
  + **Qt Quick 3D**: модуль для трехмерной визуализации.

**Логические структуры данных**

* **Переменные класса**:
  + self.vertices: массив координат вершин трехмерного объекта.
  + self.edges: массив ребер объекта.
  + self.transformation\_matrix: итоговая матрица преобразований.
* **Виджеты интерфейса**:
  + Поля ввода (QLineEdit) для координат вершин и параметров преобразований.
  + Кнопки (QPushButton) для выполнения преобразований и очистки холста.
  + Холст (QGraphicsView) для отрисовки графических элементов.

**Взаимодействие с пользователем**

* Пользователь вводит необходимые параметры в поля ввода.
* Нажимает кнопку для выполнения преобразований.
* Результаты преобразований и проекций отображаются на холсте.

**Инструкция по установке и запуску**

**Требования к системе**

* **C++** компилятор.
* Операционная система с поддержкой Qt (Windows, macOS, Linux).

**Установка**

1. **Установите Qt** с официального сайта: qt.io.
2. **Скачайте и установите Qt Creator** для удобной разработки приложений на Qt.

**Запуск программы**

1. **Откройте проект** в Qt Creator.
2. **Соберите проект** с помощью встроенного инструмента сборки.
3. **Запустите программу** из Qt Creator или командной строки.

**Инструкция пользователя**

1. **Введите координаты вершин** для создания трехмерного объекта.
2. **Выберите тип преобразования** (масштабирование, перенос, вращение).
3. **Введите параметры преобразования**.
4. **Нажмите кнопку "Применить"**, чтобы выполнить выбранное преобразование.
5. **Просмотрите результаты** преобразования и проекций на экране.

**Требования к техническим характеристикам**

* **Процессор**: с тактовой частотой не менее 1 ГГц.
* **Оперативная память**: не менее 512 МБ.
* **Дисплей**: с разрешением не менее 1024x768 пикселей.

**Обработка ошибок**

* Программа предусматривает обработку неверных вводов в полях координат и параметров. Если введены некорректные данные, преобразование не будет выполнено, и пользователь может скорректировать ввод.
* При возникновении исключений во время выполнения алгоритмов выводится сообщение об ошибке.

**Дополнительные сведения**

**Масштабирование**

* Интерфейс приложения адаптивен и корректно отображается на различных устройствах.

**Время выполнения**

* Для измерения времени выполнения алгоритмов используются встроенные функции C++.

**Сопровождение и развитие**

* Код программы хорошо структурирован и снабжен комментариями, что облегчает его поддержку и развитие.
* Для добавления новых алгоритмов преобразования и визуализации можно расширять соответствующие функции и элементы управления.

**Ссылки на используемые стандарты**

При разработке документации учтены требования следующих стандартов:

* **ГОСТ 19.101-77** — Виды программ и программных документов.
* **ГОСТ 19.402-78** — Описание программы.
* **ГОСТ 19.501-78** — Формуляр. Требования к содержанию и оформлению.
* **ГОСТ 19.505-79** — Руководство оператора.
* **ГОСТ 19.404-79** — Пояснительная записка. Требования к содержанию и оформлению.

**Заключение**

Данная программа предоставляет наглядный инструмент для изучения и демонстрации методов и алгоритмов трехмерной визуализации. Структура кода и документация соответствуют стандартам, что облегчает дальнейшее сопровождение и развитие приложения.