Введение

Актуальность(Удалить). Шейдерные программы (далее «шейдеры») – инструменты, с помощью которых можно добиться невероятных результатов в компьютерной графике. Профессиональные программисты, работающие с графикой, с помощью шейдеров создают всевозможные эффекты: добиваются эффекта фотореалистичности итогового изображения, поражают эффектами из миллиардов частиц, акцентируют внимание замыливанием заднего фона, придают плоскостям эффект объёмности и многое другое. Главной целью написание шейдеров является изменение конвейера рендеринга графики для получения желаемого результата. Современные компьютеры способны в реальном времени рассчитывать 60 раз в секунду динамические картины, близкие к максимальному фотореализму, в высоком разрешении, что ещё 20 лет назад было невозможно, и всё это благодаря шейдерам. Получив возможность менять графический конвейер, программисты смогли увеличить производительность систем построения изображений, перекладывая всё больше вычислений на графические ускорители. Ранее такой подход был невозможен, так как порядок построения изображения и функциональные возможности были строго ограничены и не могли быть изменены, из-за чего приходилось выполнять некоторые расчёты, такие как освещение, на центральном процессоре. Появление возможности писать шейдеры и встраивать их в графический конвейер изменило мир компьютерной графики в сторону гибкости и производительности. Сегодня шейдеры являются актуальной темой исследований, так как они являются важной частью любой программы, которая работает с графикой и использует для этого аппаратное ускорение, ведь благодаря им возможно быстро преобразовать геометрию, рассчитать промежуточные данные и вычислить итоговый цвет сразу для нескольких сотен пикселей в итоговом изображении. Инструментом для создания шейдера может быть любой текстовый редактор, однако не у всех есть достаточные знания для написания шейдеров, способных выдать желаемый результат. Некоторые программы стараются предоставить простой интерфейс для настройки заранее подготовленных программ, однако такой имеет ряд ограничений и не всегда оправдывает ожидания.

Главной проблемой при написании шейдеров является то, что сам процесс является нетривиальным. Одной из причин является существование разных популярных графических API, а также существование нескольких языков программирования шейдеров. Это создаёт некоторые проблемы при написании шейдеров для разных графических API, так как каждый API использует свои языки программирования шейдеров. Каким должен быть инструмент разработки шейдеров, который устранит эти недостатки?

Объект и предмет(Удалить). Объектом исследования являются шейдеры, их структура и способы их применения для решения практических задач. Предметом исследования является процесс написания шейдеров, особенности их реализации в разных API и способы упрощения разработки шейдеров.

Цель и задачи(Удалить). Целью дипломной работы является разработка программы для создания шейдеров, с помощью которой можно будет реализовать низкоуровневую и высокоуровневую логику шейдеров, используя для этого только унифицированные графические элементы. Основные задачи, которые должны быть решены для достижения цели:

* + анализ исходных данных;
  + выбор инструментальных средств для реализации;
  + проектирование программного обеспечения;
  + разработка графического интерфейса;
  + реализация функциональных частей;
  + тестирование результатов.

Предполагается, что в результате выполнения дипломной работы будет реализована программа, функциональные и графические особенности которой позволят упростить и ускорить процесс разработки шейдеров для разных API.

Методы исследования(Удалить).

основания для разработки

Труляля, подробно описаны основания для разработки!

назначение разработки

Патамушта так нада!

требования к программе или программному изделию

Ну нада, чтобы работала!

требования к программной документации

Максимально подробная

технико-экономические показатели

Блиат, хочу OpenSource!

стадии и этапы разработки

1 придумат

2 начать делат

3 закончить делат

4 закончить думат

порядок контроля и приемки

Тянуть до последнего, а потом как показать!