

большое количество функций и возможностей, которые могут не использоваться по назначению. Автоматическая генерация Mir-Map текстур является примером подобной функции. Данные текстуры призваны улучшить отрисовку объектов которые перемещаются в игре. Однако генерация данных текстур, для объектов, которые не меняют своего положения на экране, например, интерфейс игрока, не имеют смысла, так как они даже не будут использованы.

В ходе данной работы было выявлено несколько возможных причин ухудшения производительности, однако их намного больше. Для выявления данных причин необходимо хорошее понимание работы игрового приложения, процесса отрисовки изображения.

Н. В. Алексеенкова, В. В. Комраков
(ГГТУ им. П. О. Сухого, Гомель)

СОЗДАНИЕ 3D-ГОЛОВОЛОМКИ «SPACE TRAVEL: LACK OF TIME» НА UNITY

Исторически создание игр всегда являлось трудоёмким занятием, как финансово, так и физически. Но с развитием компьютерных технологий порог трудоёмкости значительно снизился. Вследствие этого возросла конкуренция и повысилась популярность игр. Однако, на данный момент разработка игр в Беларуси не очень популярна среди разработчиков. Поэтому изучение технологий по созданию игр является актуальной задачей.

Игровое приложение, о котором пойдёт речь, разработано на одном из самых популярных игровых движков Unity, который позволяет разрабатывать игры под разные платформы и операционные системы [1]. Цель игры – пройти уровни, решая головоломки, и победить в финальном сражении. На разных уровнях будут встречаться враги, передвигающиеся платформы и лабиринты, которые игроку предстоит преодолеть.

Изначально концепция игры выглядит следующим образом – данная игра является смесью жанров: платформер (от англ. Platformer), в котором основное действие заключается в преодолении препятствий передвижением по платформам, и головоломки (от англ. Puzzle), целью которой является решение логических задач, требующих от игрока задействования стратегии, логики и интуиции. Игра будет написана на

языке программирования C#, с использованием различных библиотек, поддерживаемых самой средой разработки Unity. Все элементы данной игры выполнены с использованием 3D-графики.

В заключении можно сказать, что как разработчику данная игра помогает улучшить навыки в таких сферах как разработка общей концепции игры, её сюжета, 3D-моделирование, создание игровых механик и программирование.

Литература

1 OpenGL Programming Guide. The Official Guide to Learning OpenGL, Version 4.3 / D. Shreiner [et al.]. – 8th ed. – NJ: Addison-Wesley, 2013. – 986 p.

С. А. Анисимов

(ГГУ им. Ф. Скорины, Гомель)

АЛГОРИТМ ТРАССИРОВКИ ЛУЧЕЙ ДЛЯ ОСВЕЩЕНИЯ И ОТРИСОВКИ ФИГУР В КОМПЬЮТЕРНОЙ ГРАФИКЕ

Алгоритм трассировки лучей решает проблему реалистичного освещения и отрисовки фигур в компьютерной графике. Реализация заключается в рекурсивном пускании лучей в заданном направлении и определении степени освещённости. Сначала находится ближайшая фигура. Если луч не пересёк фигуру, то он попал или в «пол» или «небо». При ситуации, когда луч пересёк «пол», то его можно пустить в другом направлении, если поверхность зеркальная. Если луч пересёк фигуру, то в зависимости от зеркальности, луч отражается в направлении нормали. В результате реализации алгоритма получается степень «яркости» пикселя. Пиксель «раскрашивается» в яркие или темные тона цвета фигуры.

Существует следующие три случая при выполнении алгоритма. Яркость отрицательна в случае расположения объекта за камерой. Отрицательные результаты рассматривать не имеет смысла, так как они не в поле видимости. При яркости в пределах от 0 до 1 объект находится между камерой и плоскостью проекции. Алгоритм реализован на языке C# в консольном приложении, где вместо цвета рассчитывается определённый символ. Результат реализации алгоритма представлен на рисунке 1.