**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение высшего образования**

**«Южно-Уральский государственный университет (национальный исследовательский университет)»**

**Институт Естественных и Точных Наук**

**Кафедра «Вычислительная механика»**

Разработка игры

(наименование темы курсовой работы (проекта)

**КУРСОВАЯ РАБОТА** (ПРОЕКТ)

**по дисциплине «\_\_\_\_\_\_\_\_\_\_\_\_\_\_Языки программирования\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_»**

**Нормоконтролер**, должность **Руководитель**, должность

##### / И.О. Фамилия / / И.О. Фамилия /

«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_ г. «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_ г.

**Автор**

**студент группы** ЕТ– 110

Д.Л.Евченко / И.О. Фамилия /

«\_11\_» \_\_\_\_июня\_\_\_\_\_\_\_\_2022\_ г.

**Работа** (проект) **защищена**

**с оценкой** (цифрой, прописью)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_ г

**Челябинск 2022**

Оглавление

[Глава 1: Постановка задачи 3](#_Toc105800966)

[Глава 2: Проектирование и разработка 9](#_Toc105800967)

[Заключение 16](#_Toc105800968)

# Глава 1: Постановка задачи

1. Название игры – «Drive or die». Название собственно означает суть игры – веди машину или умри. Жанр игры можно определять по-разному, но самое близкое – бесконечная аркадная гонка.
2. Игры из которых были позаимствованны идеи.
   1. Первая игра которая помогла мне при создании моей игры – это «Subway Surf». Subway Surf - это мобильная игра в жанре платформер, разработанная датскими компаниями Kiloo и SYBO Games. Игра насчитывает уже более миллиарда скачиваний в Google Play. Основная цель игры состоит в том, чтобы игрок, ускоряясь, пробежал как можно дальше, уворачиваясь от встречных поездов и других препятствий. С увеличением ускорения игрока — проходить препятствия становится сложнее. Если игрок с чем-то сталкивается, то инспектор его догоняет (причём вне зависимости от того, насколько далеко он успел убежать) и игра заканчивается.

Во время побега можно собирать золотые монеты и различные бонусы, благодаря которым открываются новые возможности игры.

Из этой игры я взял идею сделать несколько дорожек, по которым игрок будет перемещаться и избегать препятствий, проходя все дальше и дальше.

Также, из неё я взял монетки и бонусы, идею анимации кручения

монетки, все это делает игру интереснее и разнообразнее. 

Скриншот 1 из игры Subway Surf

 Скриншот 2 из игры Subway Surf

* 1. Вторая игра взятая в пример – это «Traffic Racer». Traffic Racer – мобильная игра в жанре бесконечная аркадная гонка, разработанная компанией Soner Cara. Суть игры – уворачиваться от столкновений и побивать рекорды, набирая все большее и большее количество очков. В игре есть внутриигровая валюта, которую игрок тратит на новые более мощные машины. Также в игре есть несколько режимов – бесконечная гонка и гонка на время.   
       
     Из этой игры я взял дорогу, а именно четыре полосы, с возможностью перемещения по всем дорогам, и рост скорости, а вместе с ней и сложности по мере продвижения.

  
Скриншот 1 из игры Traffic Racer



Скриншот 2 из игры Traffic Racer

* 1. Третья игра - «Road Fighter». Road Fighter - это [видеоигра](https://ru.wikipedia.org/wiki/%D0%92%D0%B8%D0%B4%D0%B5%D0%BE%D0%B8%D0%B3%D1%80%D0%B0) в жанре [аркадный](https://ru.wikipedia.org/wiki/%D0%90%D1%80%D0%BA%D0%B0%D0%B4%D0%B0_(%D0%B8%D0%B3%D1%80%D0%B0)) [автосимулятор](https://ru.wikipedia.org/wiki/%D0%90%D0%B2%D1%82%D0%BE%D1%81%D0%B8%D0%BC%D1%83%D0%BB%D1%8F%D1%82%D0%BE%D1%80), разработанная компанией [Konami](https://ru.wikipedia.org/wiki/Konami) и выпущенная в виде [игрового автомата](https://ru.wikipedia.org/wiki/%D0%98%D0%B3%D1%80%D0%BE%D0%B2%D0%BE%D0%B9_%D0%B0%D0%B2%D1%82%D0%BE%D0%BC%D0%B0%D1%82_(%D0%B2%D0%B8%D0%B4%D0%B5%D0%BE%D0%B8%D0%B3%D1%80%D1%8B)) 7 декабря [1984](https://ru.wikipedia.org/wiki/1984_%D0%B3%D0%BE%D0%B4_%D0%B2_%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D1%8B%D1%85_%D0%B8%D0%B3%D1%80%D0%B0%D1%85) года.

Игрок управляет гоночным автомобилем оснащенным двигателем, наблюдая за игровой ситуацией сверху. Дорога расположена вертикально, при движении фон прокручивается сверху вниз. Дорога обычно имеет небольшую ширину по отношению к ширине экрана, повороты выполнены в виде изгибов дороги в пределах экрана. Главным ограничением является запас топлива. Топливо затрачивается независимо от скорости движения, даже при полной остановке. Игра закончится, если топливо будет израсходовано полностью.

Из этой игры я взял расположение дороги и механику прокрутки фона при движении машины, а также топливо и окончание игры в случае если кончается топливо.



Скриншот из игры Road Fighter

* 1. Ну и четвертая игра которая помогла при создании – это «Crimsonland». Crimsonland - [компьютерная игра](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%B8%D0%B3%D1%80%D0%B0) в жанре [аркада](https://ru.wikipedia.org/wiki/%D0%96%D0%B0%D0%BD%D1%80%D1%8B_%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D1%8B%D1%85_%D0%B8%D0%B3%D1%80#%D0%90%D1%80%D0%BA%D0%B0%D0%B4%D0%B0) с элементами [RPG](https://ru.wikipedia.org/wiki/%D0%A0%D0%BE%D0%BB%D0%B5%D0%B2%D0%B0%D1%8F_%D0%B8%D0%B3%D1%80%D0%B0). Игрок в роли десантника-гуманоида уничтожает различных [монстров](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%BD%D1%81%D1%82%D1%80). Сюжета как такового в игре нет: в режиме Quest задача состоит в том, чтобы уничтожить всех монстров, в режиме же Rush или Typ’o’Shooter — в том, чтобы уничтожить как можно больше монстров; в режиме Survival — в том, чтобы продержаться как можно дольше.

Из этой игры я взял механику стрельбы и наличие ограниченного запаса патронов.

Ссылка на страницу игры в магазине Steam [https://store.steampowered.com/app/262830/Crimsonland/?l=russian]

Скриншот 1 из игры Crimsonland.

1. Техническое задание
2. В главном меню у игрока есть возможность начать игру, выйти из игры и изменить управление
3. При активации кнопки «Пробел» в главном меню будет проигрываться анимация выстрелов
4. После начала игры у игрока есть возможность включить паузу на кнопку Esc и после продолжить игру, либо выйти в главное меню
5. Машинка может перемещаться вправо, влево, вверх и вниз
6. Управление по умолчанию – WASD, но есть возможность переключить управление на стрелочки в главном меню
7. Машинка может стрелять пулями
8. При попадании пули в препятствие, препятствие уничтожается
9. В начале игры машина заправлена полностью, но по ходу игры топливо кончается
10. На старте игры игроку дан начальный капитал – 40 монет и 3 пули
11. По ходу игры игроку будут попадаться монеты, при столкновении с которыми увеличивается количество монет
12. При столкновении с препятствием игра завершается
13. При растрате всего топлива игра завершается
14. На иконке топлива отображается время до следующего топлива
15. Монеты можно тратить на покупку пуль, режима «Неуязвимость» и топлива, используя соответствующие клавиши – Q E F
16. Режим «Неуязвимость» активируется на 15 секунд и стреляет градом пуль, при этом их не расходуя
17. После завершения игры игроку показывается счёт и кнопка выхода в главное меню
18. В главном меню игроку отображается рекорд

# Глава 2: Проектирование и разработка

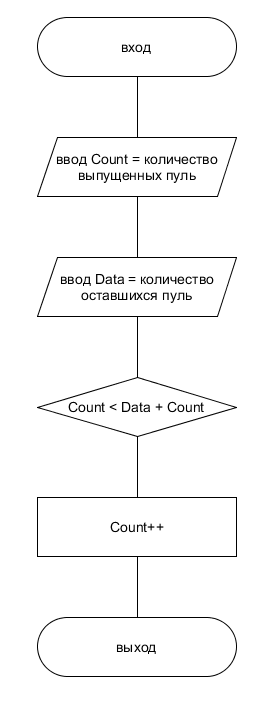
1. Игровые ресурсы
   1. Текстура машинки была взята с сайта

[https://craftpix.net/product/top-down-police-and-military-car-sprites/]

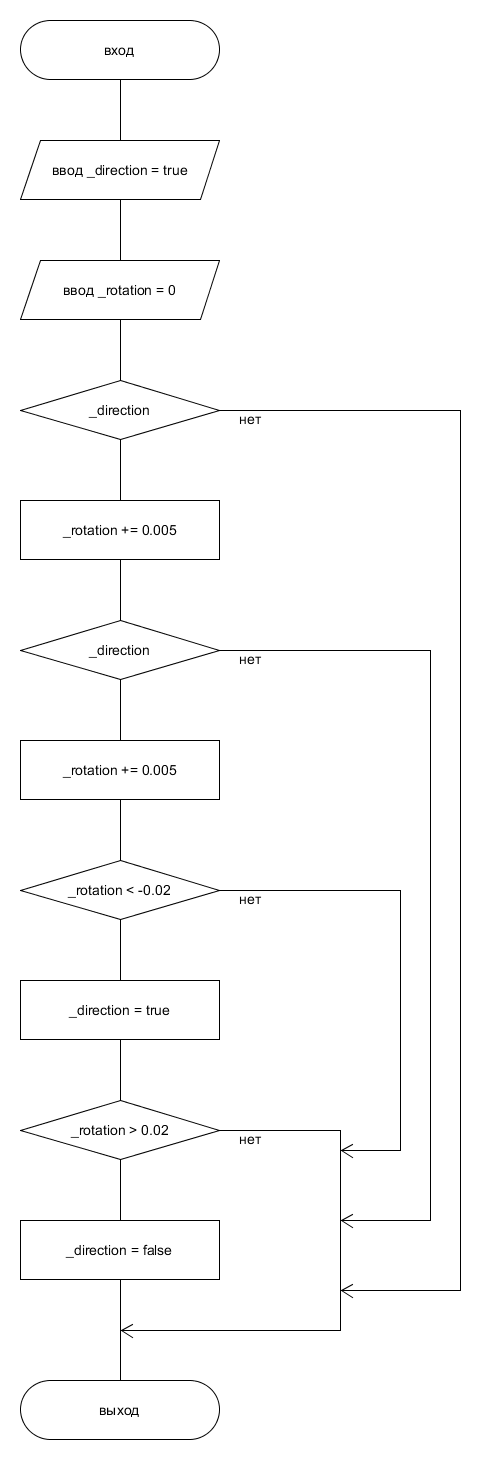
к этой текстуре я самостоятельно добавил пушку

* 1. Текстура дороги была самостоятельно нарисована в Photoshop
  2. Текстуры для меню и меню паузы были нарисованы самостоятельно также в Photoshop

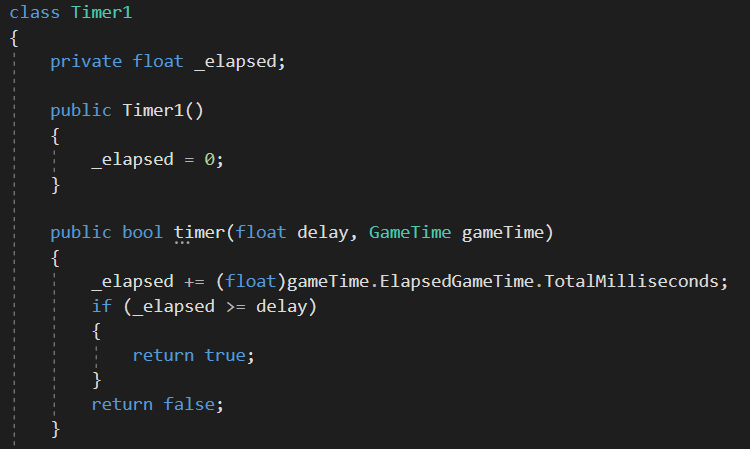
1. Основные классы
   1. GameStateManager – класс с помощью которого можно трнслировать различные сцены, в нем прописана логика переключения сцен.
   2. Data – класс в котором объявляются основные статистические поля
   3. Component – базовый класс для большинства классов, позволяет автоматически устанавливать необходимые методы
      1. Основные методы: LoadContent, Update, Draw
   4. Timer1 – класс позволяющий выполнять какое либо действие по прошествии определенного времени.
      1. Основные методы: сycleTimer – возвращает True каждые n секунд, Timer – возвращает True один раз через n секунд
   5. RandomSet – класс, хаотично расставляющий игровые объекты на дороге
   6. GameScene, MenuScene, PauseScene, GameOverScene – классы отвечающие за логику той или иной сцены.
   7. MyCar – класс, отвечающий за главный игровой объект – машину.
      1. Основные методы: Update – логика перемещения, Draw – отрисовка
      2. Основные поля: \_texCar – текстура машинки, \_posCar – позиция машинки
   8. NewGame – класс который вызывается при начале новой игры, устанавливаются начальные параметры опеределенные в классе Data
   9. KeyBoardControl – класс в котором определены взаимодействия пользователя с клавиатурой
      1. Основные поля: \_ksPrevious – прошлое состояние клавиатуры, \_ksCurrent – текущее состояние клавиатуры
      2. Основные методы: SetObj – метод в котором определено управление машинкой, ShopBuy – метод в котором определена покупка бонусов
   10. ManagerCollision – класс отвечающий за столкновения различных игровых объектов
   11. ManagerBullet – класс отвечающий за пули
       1. Основные поля: \_tex – текстура пули, \_pos – позиция пули.
       2. Основные методы: Update – метод отвечающий за позицию пули, Shoot – метод выпускающий пулю, Draw – метод отрисовывающий пулю
2. Интересные алгоритмы
   1. Алгоритм добавляющий пулю в лист отрисовок



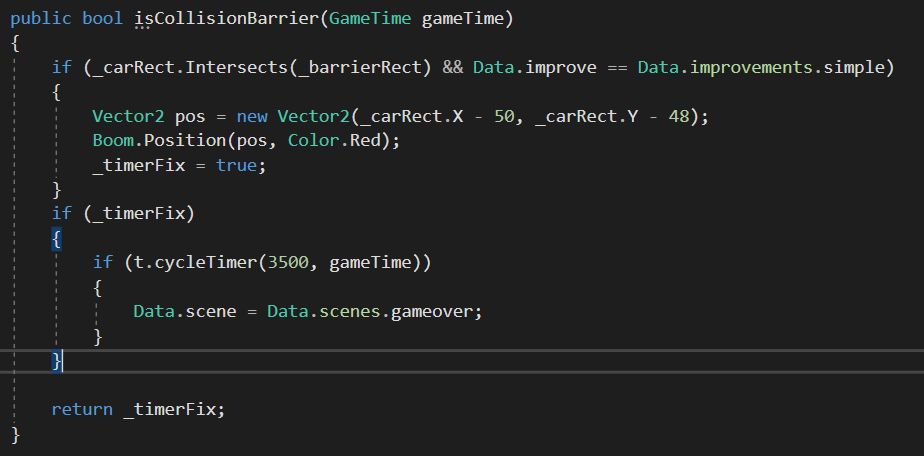
* 1. Алгоритм «покачивания» автомобиля



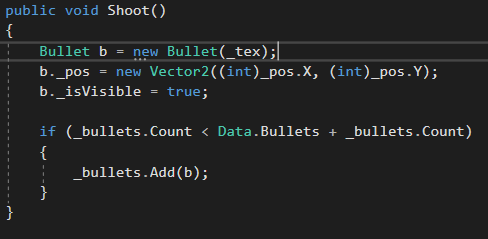
1. ЛИСТИНГ/код
   1. Класс Timer1 и метод в нем – timer, пока поле \_elapsed, к которому прибавляется некоторое значение меньше delay, возвращается false,  
      когда \_elapsed становится больше delay возвращается true

****

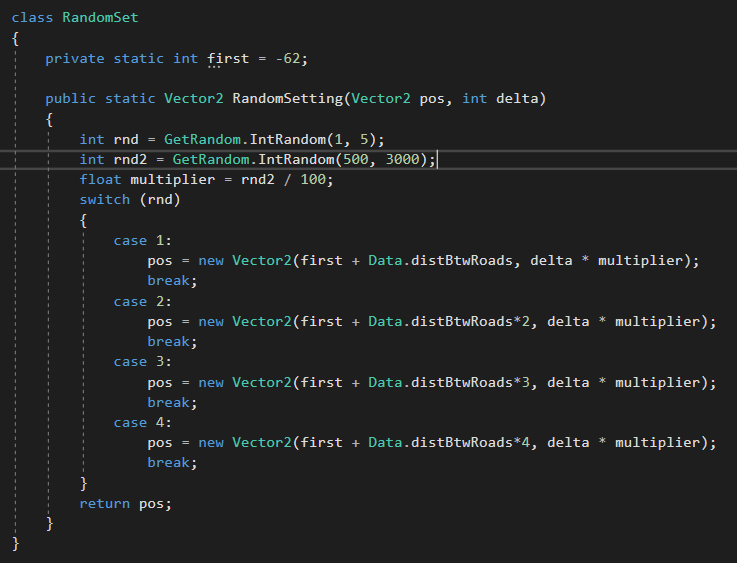
* 1. Метод isCollisionBarrier в классе ManagerCollision. Здесь осуществляется логика столкновения машины и препятствия. Поле \_timerFix, установленное по умолчанию со значением false, изменится на true когда произойдет столкновение, но перед этим проиграется анимация столкновения и после завершения анимации включится сцена gameover



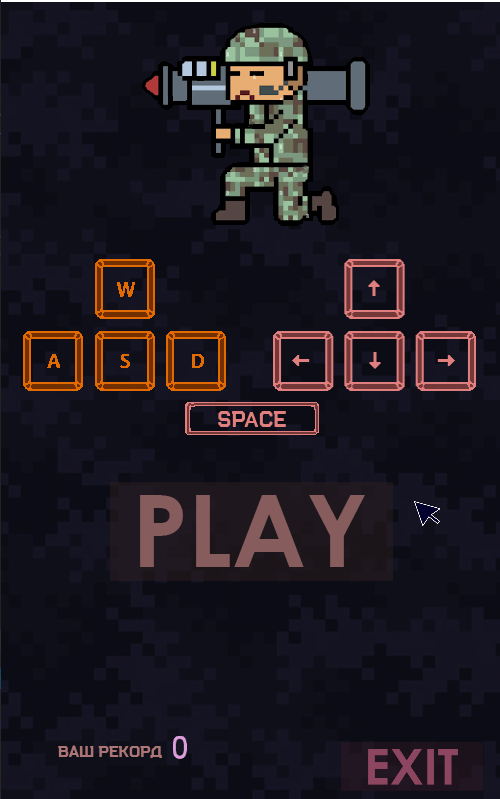
* 1. Метод Shoot из класса ManagerBullets – присваивает каждой пуле позицию автомобиля - \_pos, после этого делает ее видимой меня \_isVisible на true и добавляет в список отрисовки если текущее количество отрисованных меньше чем сумма от рисованных и оставшихся.

****

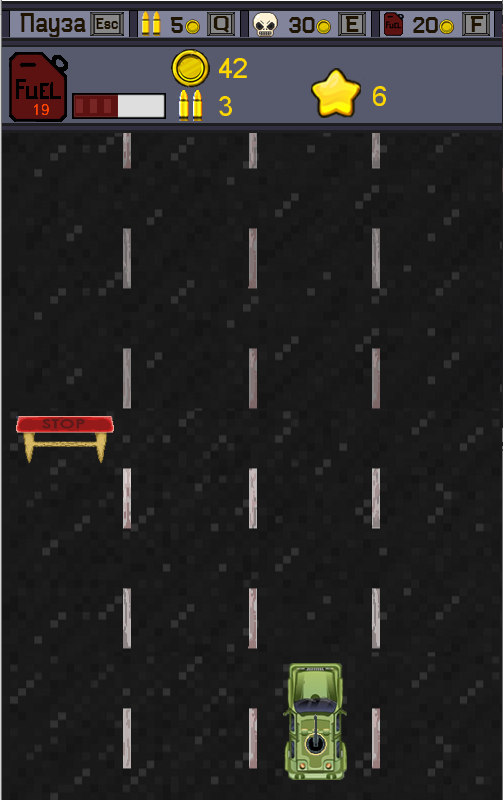
* 1. Класс RandomSet в котором происходит случайная расстановка объектов на дороге. Создается два рандомных числа – rnd (отвечающее за расположение по X) и rnd2 (отвечающее за расположение по Y). Эти числа используются для вычисления случайной позиции.



1. Особенности игры
   1. Так выглядит главное меню, при нажатии на стрелочки управление переключится на соответствующие клавиши, нажав на пробел запустится анимация выстрелов, кнопка PLAY запустит игру, ну а кнопка EXIT закроет её

****

* 1. Так выглядит сама игра, сверху изображена панель статистики, снизу управляемая игроком машинка

****

# Заключение

Разрабатывая данную игру я научился работать с базовыми принципами и инструментами MonoGame. Так же были разобраны основы объектно-ориентированного подхода к программированию.

В игре ещё остались некоторые недоработки и недочеты, но уже на данном этапе в нее можно играть.