**Проект**

**Создание игры**

**«Три в ряд»**

**с помощью Pygame**

Выполнила: Ермакова Елизавета, ученица 10 класса

Преподаватель: Шишкова Н.А.

Москва, 2018

**Содержание**

1. Введение
2. Задачи и цель проекта
3. Создание игры
4. Заключение
5. Список использованных источников

**Введение**

Игра́ — тип осмысленной непродуктивной деятельности, где мотив лежит не в ее результате, а в самом процессе.

Все люди любят игры! Кто-то играл в них в детстве, используя игрушки: куклы, машинки, а кто-то использовал подручные предметы. Палки, камни, цветы, снег – все годится для игры. Но сейчас, когда технологии так стремительно развиваются, дети (да и многие взрослые), играют в игры на электронных носителях. Игровая индустрия стремительно развивается, каждый день придумываются все новые и новые способы создания игр.

Например, Пит Шиннерс решил, что неплохо бы было создать библиотеку на Python, которая позволяла бы создавать полноценные игры. Так появилась Pygame, с помощью которой была создана моя игра.

На данный момент существуют сотни жанров компьютерных игр. Аркады, симуляторы, стратегии, шутеры, приключения…

Но моим любимым жанром являются логические игры. Они не только приносят удовольствие, но и развивают мышление.

Поэтому я решила создать игру, похожую на те, в которые я люблю играть - игру «Три в ряд».

Думаю, каждый играл в подобные игры. Данный жанр очень популярен. В подобных играх нужно собирать объекты, передвигая их так, чтобы образовывались ряды длиной 3 и более, зарабатывая очки и игровую валюту. Игра предельно проста, но и сложна одновременно.

**Задачи и цель проекта**

Цель проекта – создать игру «Три в ряд», используя Pygame.

Задачи проекта:

1. Обдумать концепцию игры
2. Собрать необходимые для приложения файлы (картинки, звуки, музыку и т.д.), а также создать необходимые файлы самостоятельно (уровни, файлы для хранения информации)
3. Написать игру на языке Python, обеспечить функционал, используя Pygame
4. Оценить получившийся проект

**Создание игры**

Прежде чем приступить к созданию игры, следует обдумать концепцию. Я хочу создать классическую игру «Три в ряд». Моя задумка – создать 10 уровней, пройдя которые вы побеждаете. Но не все так просто. Количество попыток будет ограничено жизнями. В начале игры их будет 5. Если, проходя уровень, вы не уложитесь в определенное количество ходов, то одна жизнь у вас отнимется, и после этого вы сможете попробовать пройти этот уровень еще раз. Проходя уровень, вы зарабатываете очки; очки за каждый уровень суммируются и будут выведены в конце, если вы пройдете все уровни. Также они добавляются в топ. Топ состоит из всех ваших результатов, и вы можете посмотреть, лучше вы справились или хуже, или же попытаться улучшить свой результат. В игре будет игровая валюта – монеты. За монеты вы можете покупать дополнительные жизни. Но если все жизни потрачены, увы, вам придется начать с первого уровня. Еще одна моя задумка – сделать так, чтобы можно было продолжить играть, после того как ты вышел из игры. Например, если ты играл и дошел до 4-ого уровня, ты можешь выйти из игры, а запустив её снова, ты все еще будешь на 4-ом уровне, и твои данные не потеряются. Для этого я создала несколько текстовых файлов, хранящих данные игрока:

1. All\_points – файл, который хранит текущие очки.
2. Current\_level – файл, который хранит номер текущего уровня
3. Current\_lives – файл, который хранит текущее количество жизней
4. Current\_money– файл, который хранит текущее количество монет
5. Top\_points – файл, который хранит топ очков

Также, я создала 10 файлов, в которых хранятся уровни. В файлах с уровнями было расположение объектов, количество ходов, а так же цели уровня. Создав все нужные файлы, я приступила к поиску нужных картинок и звуков. Картинку с фруктами я нашла в интернете и самостоятельно вырезала из неё все фрукты, добавив прозрачный фон. Скачав в интернете все нужные картинки, музыку и звуки для игры, и поместив их в папку data, я приступила к написанию кода.

Для спрайтов, на которые надо нажимать, я создала 4 класса (Tile, Play, Restart и Continue). Также я создала 15 функций, для разных нужд. Основной функцией стала функция game, в которой происходила сама игра. Функции start\_new и start\_screen создавали экраны начала нового уровня и начала новой игры соответственно. Функция win контролировала поведение в выигрышных ситуациях, а lose в проигрышных. Функция load\_level была создана для чтения уровня, а функция load\_image для загрузки картинки. Функция generate\_level генерировала уровень, состоящий из объектов класса Tile, используя список цифр. Функция terminate завершала работу программы. Остальные 6 функций более интересны. В большинстве своем они вызываются в функции game и служат для самого процесса игры. Об их необходимости я понимала в процессе написания кода и постепенно их добавляла. Функция change меняла местами объекты. При этом она проверяла, будет ли при этом создана комбинация. Если комбинация не будет создана после обмена – то эта функция оставит все, как было, вернув объект назад. Для неё, а так же для функции need\_to\_shuffle была создана функция is\_there\_a\_combination, проверяющая поле на наличие комбинаций. Вышеупомянутая need\_to\_shuffle была создана, чтобы проверять, можно ли вообще подвинуть что-то. Признаюсь, сначала я не задумывалась о создании данной функции, но пару раз сыграв в свою игру, я поняла, что без нее мне не справится, ведь часто случаются такие ситуации, когда ты не можешь что-то подвинуть и тогда единственный выход – это просто закрыть игру, а значит потерять одну жизнь. Need\_to shuffle проходит в цикле по всем клетками поля и двигает их во всех возможных направлениях. После этого она сообщает, была ли где-то обнаружена комбинация. Следующая функция – destruction. Она была создана для разрушения клеток, если они встали в ряды длиной 3 и более. Разрушает она их, заменяя в списке объектов цифры на пустые строки. Для того чтобы находить, сколько клеток стоит в ряд, я создала рекурсивную функцию check\_seq. Последняя, но не менее важная функция – функция falling. Она контролировала падение объектов, а также генерировала случайные объекты сверху поля.

**Заключение**

Итог моей работы – готовая игра. Мне удалось выполнить поставленные задачи и создать игру «Три в ряд» (рис.1-7).

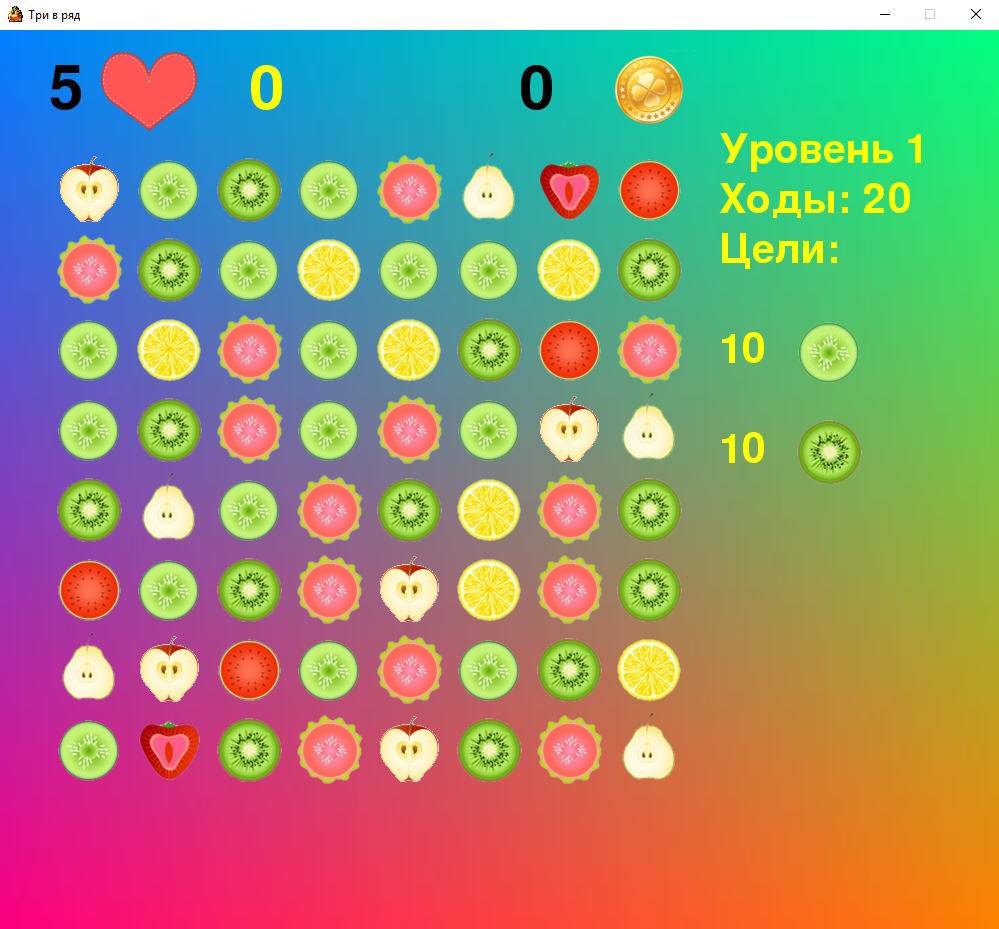
Развивая игру, можно будет добавить еще много всего, например, создать еще больше уровней, добавить усиления, которые можно покупать за монеты, и, так же, как и во многих играх этого жанра, бомбочки при сборке более 3-ех конфет в ряд.



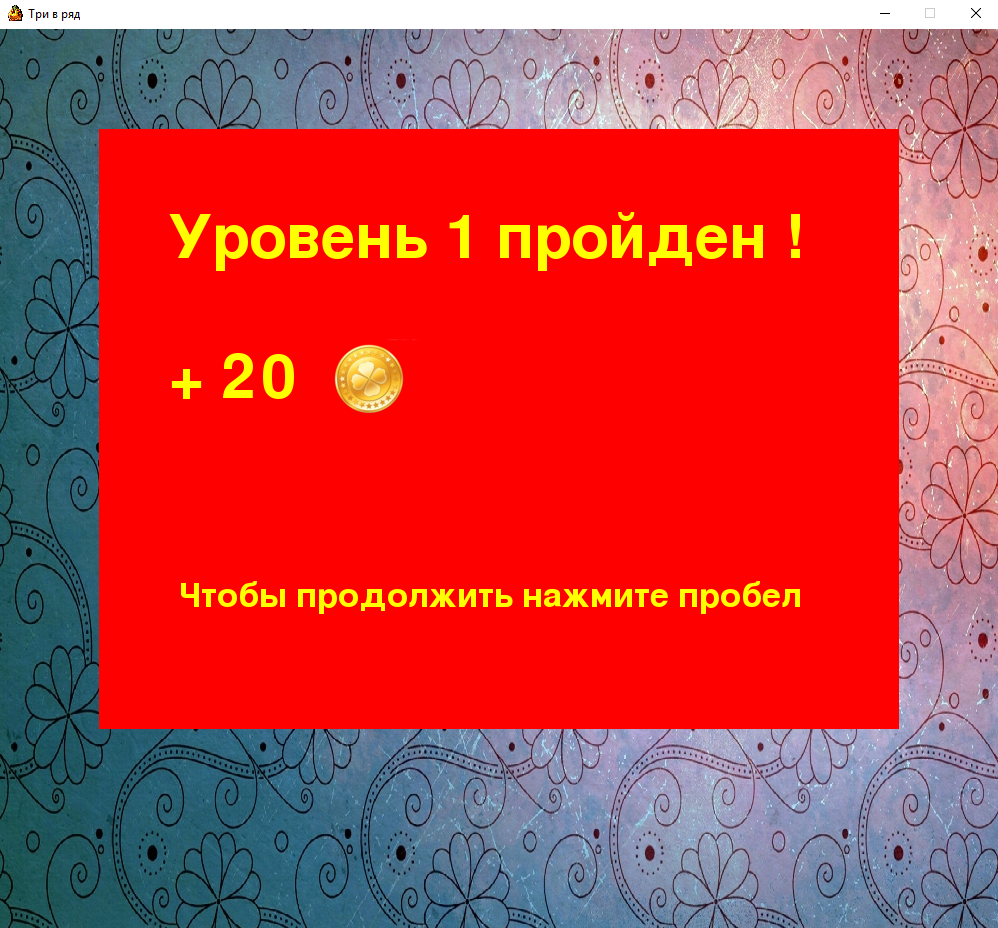
*(рис.1) Главный экран*



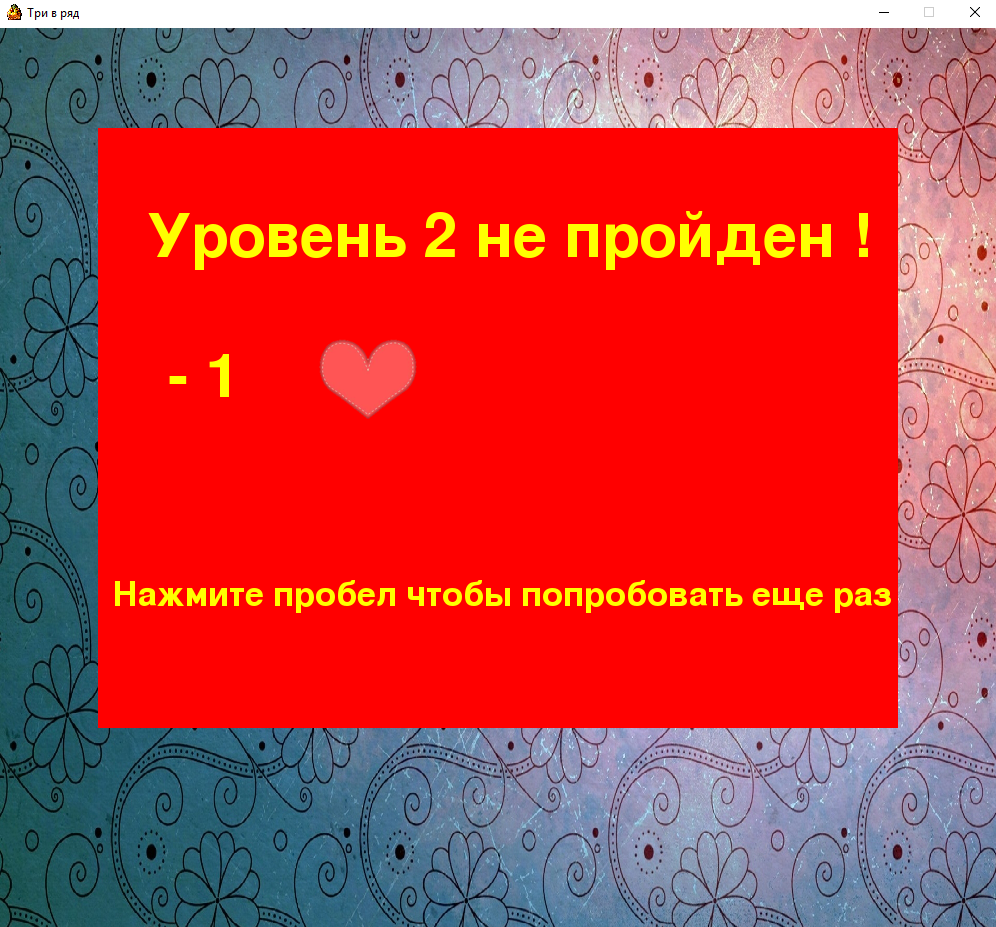
*(рис.2) Экран запуска уровня*



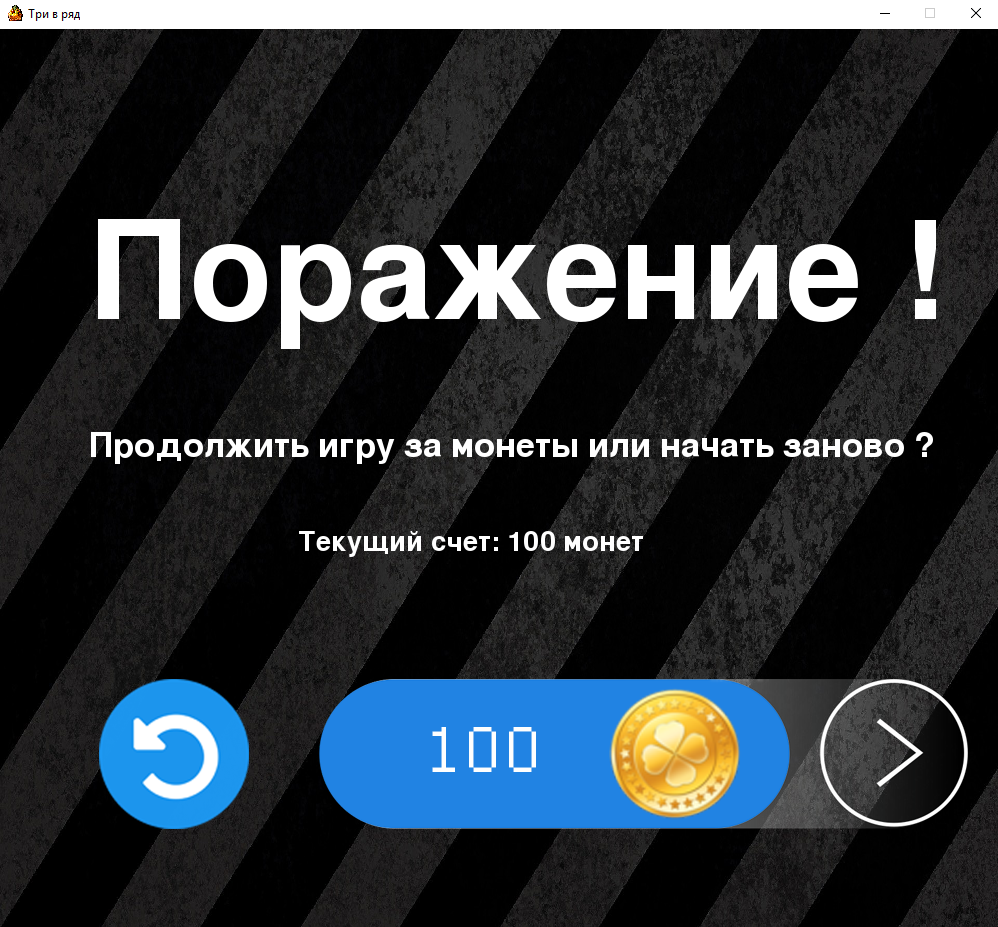
*(рис.3) Процесс игры*



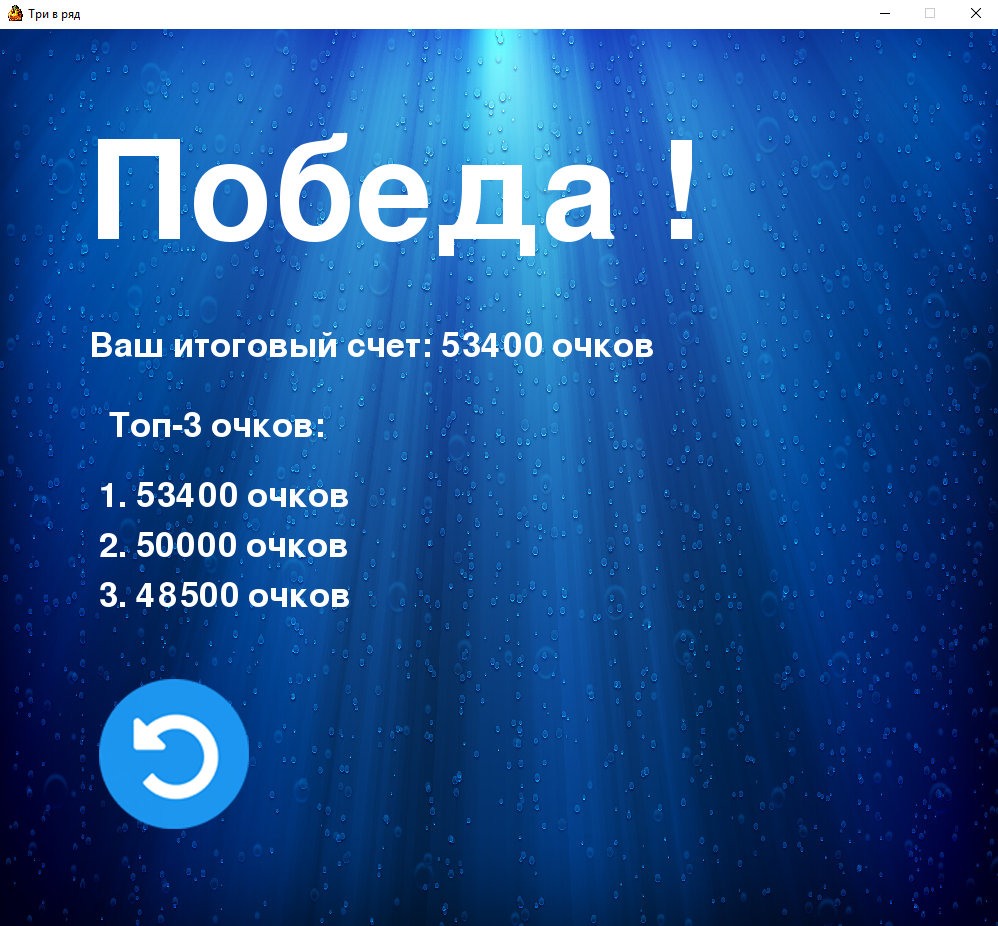
*(рис.4) Уровень пройден*



*(рис.5) Уровень не пройден*



*(рис.6) Поражение*



*(рис.7) Победа*

**Список использованных источников**

1. <https://ru.wikipedia.org>
2. <https://freesound.org>
3. <https://www.pygame.org/docs/>