**Пояснительная записка**

На разработку клона популярной игры “Марио”

1. **Цель  и общее описание:**

**Цель работы:** разработка игры-клона популярной [видео- и компьютерной игр](https://ru.wikipedia.org/wiki/Видеоигра)ы *Super Mario Bros.* на языке **Python** с использованием библиотеки **Pygame** и применением средств командной работы и контроля версий **GitHub** и СУБД **SQLite**

Разработанная игра практически полностью следует общей канве игры-оригинала в плане персонажей и сути игрового процесса, но местами иметь индивидуальные отличия. Цель разрабатываемой игры должна состоять в том, чтобы, последовательно преодолевая уровни, добраться до конца, “убив” при этом как можно более врагов и препятствий и заработав как можно больше очков.

В написанном коде предусмотрена возможность “зарабатывания” и “потери” жизни героя, набора очков для ведения рейтинга игроков. Герою “противостоят” разные виды врагов различающиеся логикой “поведения”.

Есть три уровня, разной по сложности. Эти «лабиринты» указаны в файле .txt и заданы с помощью следующих символов

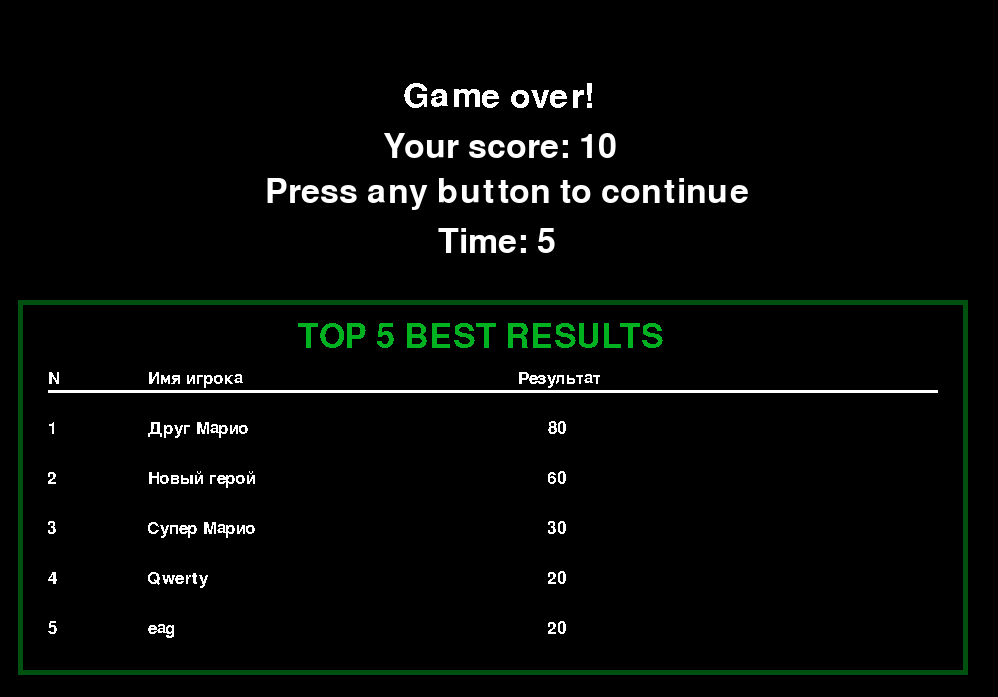
* \* - ничего (фон)
* # - препятствие
* q- секретный блок
* e - враг
* E - враг второго уровня сложности
* u - игрок
* w - конец уровня

Прохождение каждого уровня ограничено по времени — изначально на прохождение дается 150 секунд. Это время можно увеличить или уменьшить, открывая секретные боксы (для этого над прыгнуть и коснуться его снизу), либо, набрав 100 очков.

Очки используется как для добавление времени, так и для подсчета результатов разных игроков. Получить их можно, прыгнув «сверху» на врага первого уровня. За каждого «убитого» врага начисляется 10 очков. Итоговый результат заносится в базу данных. Свое имя пользователь вводит в самом начале игры, а после прохождение 3 уровней или проигрыша - видит таблицу с общими результатами. База данных, как и указано в техническом задании, содержит в себе 4 таблицы для хранения времени, результата, игрового времени и имени игрока и имеет следующую структуру:



Проигрыш может случиться по 2 причинам — уменьшения количества жизней до 0 или, если закончилось время.



Кроме основных видов клеток, описанных выше, есть еще «lucky box” – они разбросаны по полю случайным образом, и их них с разной вероятностью может выпасть добавление/вычитание жизни и времени, добавление очков и тд, или этот бокс может и вовсе оказаться пустым. При «открытии бокса» на экране появляется надпись, что в нем было

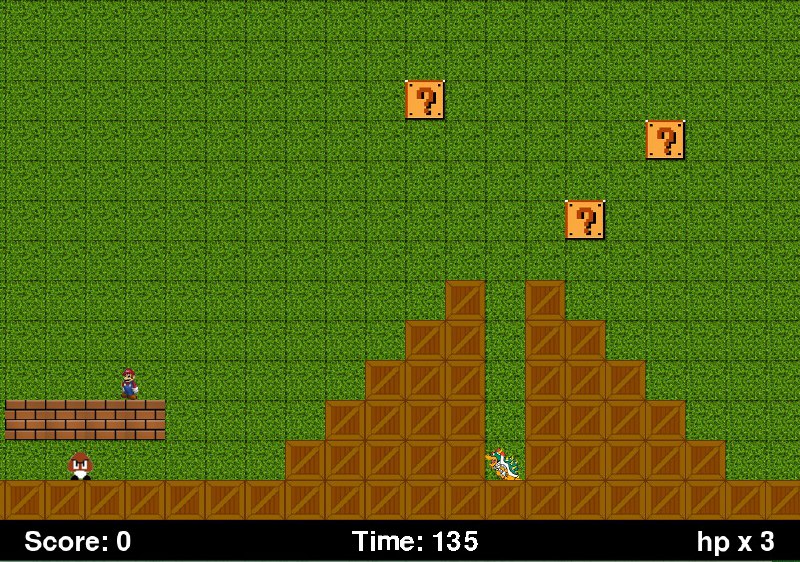


**Интерфейс программы (проект)**

совпадает с ТЗ и содержит 3 экрана:

1. Первый экран, который видит пользователь -экран приветствия. Он должен запрашивать имя пользователя (обязательное поле) и предоставляет возможность начать игру или отказаться от нее. Важно отметить, что имя пользователя вводится в элементы библиотеки PyGame (прямоугольник)



1. Второй экран - основной экран игры (размером 1000 х 700 пикселей). Помимо главной области игрового поля на нем должны отображаться текущее количество жизней, очков и остающееся время.
2. Третий экран -завершающий — показывает результат игры и топ-5 лучших результатов (с возможным учетом текущего)



1. Управление. Управление персонажем осуществляется кнопками клавиатуры. Для прыжка можно использовать пробел, стрелочку вверх, кнопку “W”; для того, чтобы пойти влево и вправо — соответствующие строчки или «D»/«A»

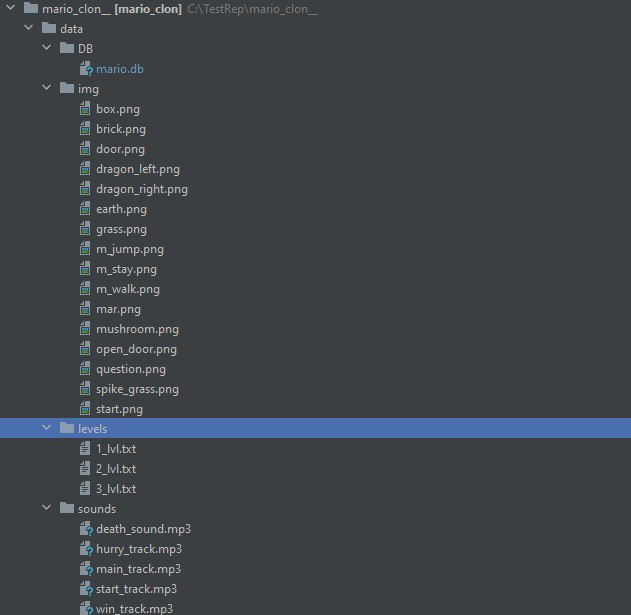
**При написании программы были учтены физические коэффициенты свободного падения (для прыжка и падения героя), использовались библиотеки PyGame, SQL,random, Time, datetime, есть взаимодействие с музыкальными, текстовыми и графическими файлами.**

**С полным списком библиотек можно ознакомится в requirements.txt**

**Хранение данных:**

Уровни, изображение и звуки расположены в разных папках, в data; к ним идет обращение из кода.

С полной структурой данных можно ознакомиться на картинке:



Все файлы загружены в на GitHub в единый репозиторий: <https://github.com/mixail612/mario_clon.git>

Все задания по ТЗ были выполнены. Дополнительно добавлены Lucky Box, с рандомными элементы и разработано 3 уровня вместо 2 изначально планируемых.