**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

На разработку клона популярной игры “Марио”

## Цель и общее описание:

**Цель работы:** разработка игры-клона популярной [видео- и компьютерной игр](https://ru.wikipedia.org/wiki/%D0%92%D0%B8%D0%B4%D0%B5%D0%BE%D0%B8%D0%B3%D1%80%D0%B0)ы *Super Mario Bros.* на языке **Python** с использованием библиотеки **Pygame** и применением средств командной работы и контроля версий **GitHub** и СУБД **SQLite**

*Super Mario Bros*. - игра в жанре [платформера](https://ru.wikipedia.org/wiki/%D0%9F%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B5%D1%80) (это жанр компьютерных игр, в которых основу процесса игры составляют прыжки по платформам, лазанье по лестницам и пр.), разработанная и выпущенная в 1985 году японской компанией [Nintendo](https://ru.wikipedia.org/wiki/Nintendo) для платформы [Famicom](https://ru.wikipedia.org/wiki/Famicom). Занесена в «[Книгу рекордов Гиннесса](https://ru.wikipedia.org/wiki/%D0%9A%D0%BD%D0%B8%D0%B3%D0%B0_%D1%80%D0%B5%D0%BA%D0%BE%D1%80%D0%B4%D0%BE%D0%B2_%D0%93%D0%B8%D0%BD%D0%BD%D0%B5%D1%81%D1%81%D0%B0)» как самая продаваемая игра в истории.

Разрабатываемая игра должна следовать общей канве игры-оригинала в плане персонажей и сути игрового процесса, но может 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 логике игры должна быть. предусмотрена возможность “зарабатывания” и “потери” жизни героя, набора очков для ведения рейтинга игроков. Герою должны “противостоять” враги не менее двух уровней, различающиеся логикой “поведения”. Также следует предусмотреть временное ограничение на прохождение каждого уровня (150 секунд) с возможностью получения дополнительных секунд за определенные достижения (например, за “победу” над 10 врагами). Логика игры должна иметь также элемент случайности - “секретный бокс”, который с определенной вероятностью может трансформироваться в дополнительную игровую жизнь/потерю жизни, дополнительное время, либо в различные препятствия.

## Структура данных

Данные для разрабатываемой игры подразделяются на

* текстовые (описание уровней)
* графические (изображения уровней, героя и врагов)
* звуковые (музыкальное “сопровождение” процессы игры и отдельных событий, например, смерти героя)
* база данных (хранение результатов прохождения игры пользователем)

#### Хранение уровней.

Разрабатываемая игра должна позволять легко изменять структуру “игрового мира”. Для этого хранение описания уровней должно осуществляться в txt файле, в котором каждый символ соответствует одной клетке игрового поля.

Отдельная строка файла задает столбец клеток поля (поле задается вертикально и транспонируется при загрузке уровня). Файл с описанием уровня должен позволять задать структуру поля с препятствиями нескольких видов, а также начальное местонахождение самого героя и врагов.

обозначения содержимого клетки:

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

Упрощенный пример текстового описания уровня:

#\*\*\*\*\*\*\*\*

#u\*\*\*\*\*\*\*

#\*\*\*\*\*q\*\*

#\*\*#\*\*\*\*\*

#\*\*#\*\*\*\*\*

#\*\*\*\*#\*\*\*

#e\*\*\*\*\*#\*

#\*\*\*\*\*\*#\*

#\*\*\*#\*\*\*\*

#\*\*E\*\*\*\*\*

#\*\*\*\*\*q\*\*

#\*\*\*#w\*\*\*

Его возможная реализация в игре:



#### Хранение результатов

Для хранения игровых результатов в базе данных достаточно одной таблицы простейшей структуры (следует сохранять имя игрока, дату и время игры, набранные баллы)



**Интерфейс программы (проект)**

1. Первый экран, который видит пользователь -экран приветствия. Он должен запрашивать имя пользователя (обязательное поле) и предоставлять возможность начать игру или отказаться от нее. В программе должны обрабатываться события от клавиатуры и мыши.
2. 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.
3. Третий экран -завершающий - должен показывать результат игры и топ-5 лучших результатов (с возможным учетом текущего)
4. Управление. Управление персонажем должно осуществляться кнопками клавиатуры. Для удобства следует предусмотреть дублирование функций управления: герой должен перемещаться как “стрелками”, так выбранными символьными клавишами, а также пробелом.

**Этапы создания проекта**

1. написание ТЗ
2. определение, какие элементы будут использовать в игре (герой, враги, препятствия), поиск/разработка нужных графических и звуковых файлов
3. создание и настройка репозитория GitHub для возможности совместной работы и контроля версий
4. создание основы игры, в которую входят: обработка загружаемых описаний уровней, физика движения врагов и персонажа (“падение” героя должно учитывать коэффициент свободного падения) и его взаимодействия с элементами на поле (препятствия и враги), переход персонажа на следующий уровень
5. создание уровней
6. создание механизма для учета жизней, очков и времени
7. создание заставки, подключение базы данных для записи результатов