Mario bros

Суть проекта

Игра Mario bros для одного пользователя

## Инструкция

При запуске приложения открывается меню с кнопками «начать игру», «опции». В правом верхнем углу показано текущее количество монет, которое увеличивается после успешного завершения уровня, и максимальное количество очков - рекорд. Во вкладке «опции» можно ознакомиться с правилами и выбрать одного из двух персонажей в магазине – марио или луиджи. Движение по опциям осуществляется с помощью клавиш W, S, а для выбора пункта – Enter. При нажатии на кнопку «начать игру» запускается первый уровень. Движение во время игры осуществляется через клавиши WAD и клавиши вверх/влево/вправо. Во время прохождения уровня можно заработать монетки (coin), задевая головой блоки с бонусами. Так же ведется счет (point). За каждого убитого врага или взятую монетку начисляются очки. Также есть возможность поставить игру на паузу или выйти в меню прямо в течение игры. Если герой сталкивается с врагами или падает в ямы – он умирает. Выводится экран с сообщением game over, уровень начинается заново по щелчку мыши. При успешном прохождении уровня и касания замка выводится меню с сообщением о победе, количеством очков, набранных за игру, рекордом, а также кнопки : «вернуться в меню», «пройти заново», «пройти следующий уровень».

## Особенности

Количество строк: 1118  
игра написана с помощью модуля pygame

## Применённые технологии

* Классы;
* Функции;
* Файл txt\* (для хранения рекорда и количества монет);

\* файл создается в ходе работы программы, если его нет.

## Структура

Классы:

* Классы подвижных героев

1. Moving(pygame.sprite.Sprite) – основной класс, от которого наследуются все враги и движущиеся элементы, кроме основных героев

Функции: play() – для обновления спрайта при каждом прохождении игрового цикла

kill\_sprite() – функция самоуничтожения, прописывается отдельно для каждого класса-наследника

1. Turtle\_Green(Moving) – враг, зеленая черепаха
2. Mushrooms(Moving) – враг, двигающийся гриб
3. Mario(pygame.sprite.Sprite) – главный герой

Функции:

get\_event() – обработка событий нажатия клавиш

update() – обновление спрайта при каждом прохождении игрового цикла

die() – самоуничтожение при столкновении с врагами или падениями в яму

jump() – прыжки и падения

run() – бег вправо/влево

stand() – герой неподвижен

* Неподвижные спрайты

1)Base(pygame.sprite.Sprite) – базовый класс для неподвижных блоков и декораций

2) Cloud(Base) – маленькое облако

3)Big\_Cloud(Base) – большое облако

4) Brick(Base) – кирпич

5) Ground(Base) – земля

6) Bush(Base) – маленький куст

7) Big\_Bush(Base) – большой куст

8)Hill(Base) - маленький холм

9) Big\_Hill(Base) – большой холм

10)Bonus(Base) – блок с бонусом

Функции: play() – обновление картинки (мигание)

11) Tube\_Small(Base) – маленькая труба

12)Tube\_Sr(Base) – средняя труба

13) Tube\_Big(Base) – большая труба

14) Block(Base) – блоки

15) Castle(Base) – замок

16) None\_Tube(Base) – для труб

* Кнопки:

1)Button(pygame.sprite.Sprite) – базовый класс для кнопок

Функции: clicked() – прописывается отдельно для каждой кнопки

2) Pause(Button) – кнопка паузы

3) In\_Menu(Button) – кнопка возврата в меню

4) Buttons\_Wins() – базовый класс кнопок в сообщении выигрыша

5) Start\_Game\_1(Buttons\_Wins) – кнопка начала игры

6) ReGame(Buttons\_Wins) – прохождение уровня заново

* Класс Camera()
* Класс Map()

Функции:

generate\_level() – создание уровня

make\_tube() – создание трубы

* Класс OpenMenu(pygame.sprite.Sprite) – кнопка открытия меню опций

Функции:

1. rules - правила
2. choose\_hero – выбор персонажа
3. update

* Menu() – класс панели меню

Функции:

append\_option

switch

select

draw

* LuidziChose(pygame.sprite.Sprite) –выбор Луижди
* MarioChose(pygame.sprite.Sprite) –выбор Марио
* Start\_Game – начало игры
* EndGame – конец игры, вывод окна с сообщением о победе

Функции:

next\_lev() – переход на следующий уровень

data\_result – запись в txt файл рекорда

read\_result – прочтение из txt рекорда

start() – начало!!!

game\_over() – проигрыш

start\_game() – начало игры

up\_mask() – проверка для пересечения по маске при столкновении марио с врагами

start\_sprite() – начало движения спрайтов, которые изначально были вне поля зрения камеры

collidepoint\_by\_mask() – пересечение по маске

get\_nearest\_block() – ближайшие блоки сверху/снизу/справа/слева

flip() – поворот всех фреймов

get\_offset() – отступы маски спрайта от картинки спрайта сверху/снизу/справа/слева

cut\_sheet() – обрезка фреймов

load\_image() – загрузка изображений и изменение масштаба

points\_and\_coins() – вывод очков и монет на экран в игре

get\_paused() – пауза

terminate() – выход

text() – загрузка текста