**Название проекта:**

Игра «Космический шутер».

**Ссылка:**

<https://github.com/hertysss/space-shooter>

**Авторы проекта:**

Лобанова Анна и Кананадзе Никита.

**Сфера ответственности:**

* **Разработчик:** Лобанова Анна.

**Классы:** Game, Gun, Bullet, Stats, Score, Ino

**Функции:** process\_events, update\_display, start\_game

**Коммиты:**

* + Commits on Dec 4, 2022: main, controls, Gun 51fe8f8a91854cd5bfea5b055b1091ed24fae26
  + Commits on Dec 6, 2022: добавлены классы Bullet, Game 0b49b396236b6361a72a7dae2da8db5be1ff23f8
  + Commits on Dec 10, 2022: добавлен класс Ino 4cdeafaea215da6bea88d61f39dd70126f8abc5b
  + Commits on Dec 15, 2022: база a96582338934f739622fb766c86594b4e43a706b
  + Commits on Dec 18, 2022: почти 8fbeac6252dbcd2c47d74b056ce147ffb99cc8d4
  + Commits on Dec 20, 2022: новая ветка 48828895a492ee5360d7938d32880a92f5e1ffc7
  + Commits on Dec 20, 2022: создается армия пришельцев 0af604ae9f7bed52a4d590e666dcf8135e89a1f2
  + Commits on Jan 15, 2023: delete 7357449b1cf658f06f5429c62ce4d0708801eea3
* **Разработчик**: Кананадзе Никита.

**Классы:** Creator, End

**Функции:** main\_menu, lvllist, skin\_change, start\_lvl, charupgarde, ship\_upgrade

**Коммиты:**

* Commits on Dec 20, 2022: Add files via upload 4572eb59ef88e999e465d7b5ae125c22ce858172
* Add files via upload ba11c1e14fb57c75873034943d60f32673dd1f38

**Описание программы:**

Игра представляет собой 2D shooter. После старта программы пользователь попадает в главное меню, в котором он может выбрать вид корабля и уровни сложности игры. Уровни сложности игры отличаются скоростью пришельцев, космического корабля и фоном игры. После начала игры управляя космическим кораблем стрелками вправо и влево и стреляя пушкой – пробелом нужно уничтожить армию пришельцев. Если пришельцы на текущем уровне уничтожены, то проверяется пройдены ли все уровни, и если да, то игрок попадает в конечное меню, иначе игра продолжается и кол-во рядов пришельцев увеличивается. Если в процессе игры пришельцы заходят за нижнюю границу экрана или сталкиваются с космическим кораблем, то игра останавливается, далее идет проверка есть ли еще жизни, если да, то уровень перезапускается, кол-во жизней уменьшается, иначе игрок попадает в конечное меню. В конечном меню пользователь может выйти из игры или попасть в главное меню. За каждого уничтоженного пришельца в течении игры игроку начисляются очки, отображаются в текущей игре, если рекорд побит, то рекорд сохраняется на новый. Информацию о текущем рекорде можно посмотреть в главном меню или в запущенной игре.

**Структура программы:**

**Разработчик: Лобанова Анна**

**Класс Game:**

Нужен для связки объектов других классов. При инициализации создается пушка (объект класс Gun), статистика (объект класса Stats), счетчик (объект класса Score) и армия пришельцев (группа объектов класса Ino). Класс Game проверяет не вышел ли какой-нибудь пришелец за границу экрана (метод check\_inos), проверяет нужно ли перезапустить игру (метод check\_on\_restart), отображает количество жизней игрока (метод show\_life), обновляет игру (метод update), перезапускает игру (метод restart\_game), создает армию пришельцев (метод create\_army).

**Класс Gun:**

Нужен для работы с пушкой. Класс Gun отрисовывает пушку (метод update), изменяет ее позицию (метод move), реализует стрельбу (метод shoot) пулями (объектами класса Bullet).

**Класс Ino:**

Нужен для работы с пришельцами. Класс Ino отрисовывает пришельца (метод update), изменяет его позицию (метод move).

**Класс Bullet:**

Нужен для работы с пулями. Класс Bullet отрисовывает пулю (метод update), изменяет ее позицию (метод move).

**Класс Score:**

Нужен для работы с очками в текущей игре. Класс Score отрисовывает текущие и рекордное кол-во очков (методы image\_score, image\_high\_score соответственно) и обновляет их (метод render).

**Класс Stats:**

Нужен для ведения статистики игр. Класс Score обновляет рекордное кол-во очков в json файле (метод update\_high\_score).

**Функция process\_events:**

Нужна для обработки событий с клавиатуры.

**Функция update\_display:**

Нужна для отрисовки экрана игры.

**Функция start\_game:**

Нужна для инициализации игры.

**Разработчик: Кананадзе Никита.**

**Класс End:**

Нужен для запуска конечного меню, в котором отображается собственный рекорд в случае проигрыша, новый рекорд в случае выигрыша. Так же имеет возможность возврата к меню (метод open\_menu)

**Класс Creator:**

Создает кнопку, получая в качестве аргумента надпись, callback и шрифт. (метод add\_option) Так же рисует ее, предусматривает выделения активной кнопки, перемещения между ними, активацию кнопок и их удаление. (метод switch\_point, draw, select, clearing)

**Функция main\_menu:**

Связывает ниже упомянутые функции. Кнопки в меню являются переключателями между режимами меню.

**Функция lvllist:**

При включении режима lvllisting в основном цикле запускается счетчик, от значения которого зависит то, фон какого уровня будет видеть пользователь. Кнопки в меню являются триггерами для старта соответствующего уровня.

**Функция skin\_change:**

При включении режима skin\_changing возможность переключаться между кнопками меню отключается и теперь появляется возможность полистать варианты облика корабля, а при их сохранении записывает выбранный облик в json с информацией о персонаже

**Функция start\_lvl:**

Обращаясь к функции start\_game, запускает игру.

**Функция charupgarde:**

Вызывается при попытке пользователя улучшить персонажа. При достижении нужного рекорда на соответствующем уровне позволяет пользователю улучшить корабль. В противном случае говорит ему о том, что нужно набить больше очков на конкретном уровне.

**ship\_upgrade**:

Выгружает информацию о корабле из нужного json и отображает их в меню.