**Слайд 1**

Игра «Морской бой», проект PyGame, автор Егор Корчагин.

**Слайд 2**

* Одним из самых любимых развлечений моих сверстников в СССР была игра «Морской бой», посвящённая битвам военных судов.
* Для того, чтобы скрасить скучный урок или лекцию, достаточно было взять только листок бумаги в клетку и карандаш. По мнению моих родителей, время пролетало невероятно быстро и интересно, покруче многих компьютерных игр.

**Слайд 3**

* Позднее на прилавках магазинов появились красочные игровые наборы.
* «Морской бой» развивает логическое мышление, улучшает внимание, память, приучает к усидчивости и терпению. Учит вырабатывать стратегию поведения и принятию решений.

**Слайд 4**

Я решил сделать свою цифровую версию игры, используя библиотеку Pygame, сохраняя её бумажную стилистику.

**Слайд 5 - 6**

Напомню правила

**«Морской бой»** — игра для двух участников, в которой игроки по очереди называют координаты на неизвестной им карте соперника. Если у соперника по этим координатам имеется корабль (координаты заняты), то корабль или его часть «разрушается», а попавший получает право сделать ещё один ход. Цель игрока — первым потопить все корабли противника.

Игрок рисует на листке два квадрата, размер каждого их них 10 на 10, то есть по 10 клеток по вертикали и горизонтали. Вертикаль подписываем буквами алфавита сверху вниз. По горизонтали под каждой клеткой записываем соответствующую цифру.

Игрок располагает корабли так, чтобы они не касались друг друга и между ними был зазор не менее 1 клетки.

**Слайд 7**

Перед тем, как начать играть, участник выбирает режим игры: «Робот», если он один, или «2 игрока», если участников двое.

**Слайд 8**

Каждый участник расставляет свои корабли по очереди, робот генерирует для себя случайную расстановку.

Можно перетаскивать корабли на места и поворачивать их вручную или воспользоваться функцией авто-расстановки, которая расставит корабли за пользователя.

Корабль, который был последний раз тронут пользователем отмечен красным прямоугольником.

Чтобы вернуться назад в меню, нужно нажать на кнопку «Назад» или Ctrl + B.

Чтобы повернуть корабль, нужно нажать на кнопку «Поворот» или Ctrl + R.

Чтобы расставить корабли автоматически, нужно нажать на кнопку «Авто» или Ctrl + A.

После того, как корабли расставлены, нужно нажать на кнопку «Далее» или Ctrl + N.

**Слайд 9**

Главная сцена игры – игровое поле.

По-прежнему, чтобы вернуться назад в меню, нужно нажать на кнопку «Назад» или Ctrl + B.

Наверху отображается игровая статистика: сколько кораблей осталось на каждом поле и сколько выстрелов было сделано.

Внизу подписаны поля участников.

Между полями находится зелёная стрелка, означающая чей сейчас ход.

Попадание в корабль обозначается анимацией огня, а промаха – анимацией снаряда.

**Слайд 10**

После того, как один участник потопил все корабля противника, высвечивается результат игры и статистика побед каждого участника.

**Слайд 11**

Давайте посмотрим демонстрационное видео.

**Слайд 12 - 13**

Хочу ознакомить вас с основными функциями игры:

* load\_image() – загрузка изображения
* load\_sound() – загрузка музыки

class Game:

* game\_loop() – игровой цикл
* drawing\_sprites() - рисование спрайтов
* change\_statistic() – изменение статистики
* robot\_move() – ход робота
* player\_move() – ход игрока

class GameField:

* generate\_random\_field() – авто-расстановка кораблей
* make\_move() – выполнение хода

class Ship:

* correct\_coords() - проверка правильности координат корабля
* rotate() – поворот корабля

class Backgound:

* draw\_background() – прорисовка заднего фона
* draw\_field() – прорисовка поля
* begin\_scene() – прорисовка меню игры
* placement\_of\_ships\_scene() – прорисовка сцены расстановки кораблей
* battle\_scene() – прорисовка сцены боя
* result\_scene() – прорисовка сцены результатов

**Слайд 14**

Спасибо за внимание!