

Документация к игре “2048” на Python

Авторы: Чаплыгин Алексей, Маркус Филипп, Улитин Максим\n**Учебная группа:**

ИД24-1 Цель: Предоставить нетехническое объяснение работы каждой функции в коде игры “2048”, написанной на Python с использованием библиотеки `Pygame`.

Введение

Этот документ описывает логику работы программного кода, который реализует популярную игру “2048”. Код разделен на три основные части: **Игровая логика**, **Класс для кнопок** и **Основная программа**.

1. Игровая логика

Этот раздел содержит функции, которые управляют правилами игры, движением плиток и проверкой условий победы/поражения.

Название функции	Что делает (Простое объяснение)
<code>load_high_score()</code>	Загружает рекорд. Проверяет, есть ли файл с лучшим результатом, и считывает его. Если файла нет или он поврежден, возвращает 0.
<code>save_high_score(score)</code>	Сохраняет рекорд. Записывает текущий лучший результат в специальный файл, чтобы он сохранился между запусками игры.
<code>init_board(size)</code>	Начинает новую игру. Создает пустое игровое поле (сетку) заданного размера (например, 4x4) и сразу добавляет на него две стартовые плитки.
<code>add_new_tile(board, size)</code>	Добавляет новую плитку. Находит случайную пустую клетку на поле и помещает туда новую плитку со значением 2 (в 90% случаев) или 4 (в 10% случаев).
<code>compress(board, size)</code>	Сдвигает плитки. Перемещает все плитки в один край (в данном случае, влево), чтобы между ними не было пустых клеток. Важно: эта функция не объединяет плитки, а только сдвигает их.
<code>merge(board, score, size)</code>	Объединяет плитки. Проходит по рядам и объединяет две одинаковые соседние плитки (например, две “2” становятся “4”). При объединении обновляется счет игрока.
<code>reverse(board, size)</code>	Отражает поле по горизонтали. Меняет местами левую и правую стороны поля. Используется для реализации движения вправо.
<code>transpose(board, size)</code>	Отражает поле по диагонали. Меняет местами строки и столбцы. Используется для реализации движений вверх и вниз.
<code>move_left(board, score, size)</code>	Полное движение влево. Выполняет полную последовательность действий для хода влево: сначала сдвигает плитки (<code>compress</code>), затем объединяет их (<code>merge</code>), и снова сдвигает (<code>compress</code>), чтобы заполнить образовавшиеся пустоты.

Название функции	Что делает (Простое объяснение)
<code>move_right(board, score, size)</code>	Полное движение вправо. Использует функцию <code>reverse</code> для отражения поля, затем выполняет <code>move_left</code> , и снова отражает поле обратно.
<code>move_up(board, score, size)</code>	Полное движение вверх. Использует функцию <code>transpose</code> для поворота поля, выполняет <code>move_left</code> , и снова поворачивает поле обратно.
<code>move_down(board, score, size)</code>	Полное движение вниз. Использует функцию <code>transpose</code> для поворота поля, выполняет <code>move_right</code> , и снова поворачивает поле обратно.
<code>check_game_over(board, size)</code>	Проверяет конец игры. Определяет, закончилась ли игра. Это происходит, если на поле нет пустых клеток И нет двух соседних плиток с одинаковым значением, которые можно было бы объединить.

2. Класс `Button` (Кнопка)

Этот класс используется для создания интерактивных кнопок в меню и на игровом экране.

Название метода	Что делает (Простое объяснение)
<code>__init__(...)</code>	Создает кнопку. Определяет ее положение, размер, текст, а также действие (<code>callback</code>), которое должно произойти при нажатии.
<code>draw(screen, font, theme_colors)</code>	Рисует кнопку. Отображает кнопку на экране, используя выбранные цвета темы. Если кнопка выбрана (например, в меню), ее цвет меняется.
<code>handle_event(event)</code>	Обрабатывает нажатие. Отслеживает, нажал ли пользователь мышью на область кнопки, и если да, то запускает связанное с ней действие.

3. Основная программа

Эти функции управляют запуском и отображением игры и меню.

Название функции	Что делает (Простое объяснение)
<code>run_game(grid_size_choice, theme_choice)</code>	<p>Запускает игровой процесс. Это главный цикл игры. Он выполняет следующие действия:</p> <ol style="list-style-type: none"> 1. Устанавливает размер поля и тему, выбранные в меню. 2. Инициализирует игру (<code>init_board</code>) и загружает рекорд. 3. В цикле: ждет нажатия клавиш (стрелки для хода, <code>ESC</code> для выхода в меню, <code>R</code> для перезапуска), обновляет игровое поле, добавляет новые плитки, проверяет конец игры и перерисовывает весь экран (плитки, счет, фон).
<code>main_menu()</code>	<p>Запускает главное меню. Это первая функция, которая запускается при старте программы. Она:</p> <ol style="list-style-type: none"> 1. Создает окно меню. 2. Отображает кнопки для выбора размера поля (4x4, 5x5) и темы (светлая, темная). 3. Ждет, пока пользователь нажмет кнопку “START GAME” . 4. После нажатия запускает функцию <code>run_game</code> с выбранными параметрами.
<code>if __name__ == "__main__":</code>	<p>Точка входа. Стандартная конструкция в Python, которая гарантирует, что функция <code>main_menu()</code> будет запущена только тогда, когда файл запускается напрямую, а не импортируется как модуль.</p>