

# 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>	<b>Загружает рекорд.</b> Проверяет, есть ли файл с лучшим результатом, и считывает его. Если файла нет или он поврежден, возвращает 0.
<code>save_high_score(score)</code>	<b>Сохраняет рекорд.</b> Записывает текущий лучший результат в специальный файл, чтобы он сохранился между запусками игры.
<code>init_board(size)</code>	<b>Начинает новую игру.</b> Создает пустое игровое поле (сетку) заданного размера (например, 4x4) и сразу добавляет на него две стартовые плитки.
<code>add_new_tile(board, size)</code>	<b>Добавляет новую плитку.</b> Находит случайную пустую клетку на поле и помещает туда новую плитку со значением <b>2</b> (в 90% случаев) или <b>4</b> (в 10% случаев).
<code>compress(board, size)</code>	<b>Сдвигает плитки.</b> Перемещает все плитки в один край (в данном случае, влево), чтобы между ними не было пустых клеток. <b>Важно:</b> эта функция не объединяет плитки, а только сдвигает их.
<code>merge(board, score, size)</code>	<b>Объединяет плитки.</b> Проходит по рядам и объединяет две одинаковые соседние плитки (например, две “2” становятся “4”). При объединении обновляется счет игрока.
<code>reverse(board, size)</code>	<b>Отражает поле по горизонтали.</b> Меняет местами левую и правую стороны поля. Используется для реализации движения вправо.
<code>transpose(board, size)</code>	<b>Отражает поле по диагонали.</b> Меняет местами строки и столбцы. Используется для реализации движений вверх и вниз.
<code>move_left(board, score, size)</code>	<b>Полное движение влево.</b> 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>	<b>Полное движение вправо.</b> 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>	<b>Полное движение вверх.</b> 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>	<b>Полное движение вниз.</b> 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>	<b>Проверяет конец игры.</b> Определяет, закончилась ли игра. Это происходит, если на поле нет пустых клеток <b>И</b> 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>	<b>Создает кнопку.</b> Определяет ее положение, размер, текст, а также действие ( <code>callback</code> ), которое должно произойти при нажатии.
<code>draw(screen, font, theme_colors)</code>	<b>Рисует кнопку.</b> Отображает кнопку на экране, используя выбранные цвета темы. Если кнопка выбрана (например, в меню), ее цвет меняется.
<code>handle_event(event)</code>	<b>Обрабатывает нажатие.</b> 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)
<pre>run_game(grid_size_choice, theme_choice)</pre>	<p><b>Запускает игровой процесс.</b> Это главный цикл игры. Он выполняет следующие действия:</p> <ol style="list-style-type: none"> <li>1. Устанавливает размер поля и тему, выбранные в меню.</li> <li>2. Инициализирует игру ( <code>init_board</code> ) и загружает рекорд.</li> <li>3. <b>В цикле:</b> 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).</li> </ol>
<pre>main_menu()</pre>	<p><b>Запускает главное меню.</b> Это первая функция, которая запускается при старте программы. Она:</p> <ol style="list-style-type: none"> <li>1. Создает окно меню.</li> <li>2. Отображает кнопки для выбора размера поля (4x4, 5x5) и темы (светлая, темная).</li> <li>3. Ждет, пока пользователь нажмет кнопку “START GAME” .</li> <li>4. После нажатия запускает функцию <code>run_game</code> с выбранными параметрами.</li> </ol>
<pre>if __name__ == "__main__":</pre>	<p><b>Точка входа.</b> 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>