ФГБОУ ВО “Чувашский государственный университет им.

И.Н. Ульянова”

Кафедра вычислительной техники

Лабораторная работа №4

Выполнил: студент гр. ИВТ-41-23

Сорокин А.В.

Проверила: Первова Н.В.

Чебоксары 2025

1. Цель работы:

Изучить принципы разработки динамических web-страниц с использованием клиентских сценариев JavaScript и объектной модели документа. Научиться создавать кроссбраузерные html-страницы. Познакомиться с уровнями и особенностями DOM API и обработкой событий DOM различных уровней.

1. Сценарий и интерфейс:

Игра представляет собой классический тетрис, стилизованный под портативную консоль 90-ых годов. Цель игры – выкладывая на игровом поле фигуры «тетромино», зарабатывать очки за заполненные линии, которые удаляются по мере заполнения.

1. Описание интерфейса состояний:

|  |  |  |  |
| --- | --- | --- | --- |
| Игровое состояние / элемент интерфейса | Внутренняя модель (JS-объекты/данные) | HTML-элемент (DOM) | Назначение / отображение |
| Игровое поле (playfield) | playFieldMask[x][y] — двумерный массив состояния клеток (0/1). curTet — текущая фигура. | <div class="playfield"> → множество <div class="cell" data-index="xy"> | Основное поле 10×20. Каждая ячейка отображает текущее состояние: пустая или занятая тетромино (через класс "tet"). |
| Следующая фигура (nextfield) | TetPul.next — следующий тетромино. | <div class="next"> → сетка 4×4 <div class="cell" data-index="xy"> | Показывает превью следующей фигуры для игрока. |
| Текущая фигура | TetPul.current (наследник Tetromino, например T\_type, O\_type и т. д.). | Те же <div class="cell[data-index]> внутри .playfield, которым добавлен класс "tet". | Отрисовывается на поле путём добавления/удаления класса "tet" у нужных ячеек. |
| Очки (счёт) | Объект Score, метод add() обновляет значение. | <div id="score"> внутри .menu .info | Отображает текущий счёт игрока (до 9999), обновляется при удалении линий. |
| Управление клавишами | Слушатели keydown / keyup — перемещение/поворот curTet. | — (нет отдельных DOM-элементов, только события на window) | Игрок управляет фигурой: движение влево/вправо, поворот, ускорение падения. |

1. Описание игрового сценария:

Весь игровой процесс происходит в функции Play(), которая инициализирует два ключевых объекта:

* Тикер – вызывает кол-бек функцию с определённым тик-рейтом.
* Тетромино пул – объект, генерирующий и хранящий случайный набор тетромино, которые будут добавлены на игровое поле.

Затем рекурсивно запускается игровой цикл, продолжающийся до момента, пока новую фигуру добавить на поле будет не возможно.

Каждая фигура является объектом, хранящая свою позицию, относительные положения блоков фигуры и возможность движения дальше.

После появления фигуры на поле, она каждый фиксированный отрезок времени смещается вниз до тех пор, пока это возможно. В это время перехватываются события нажатия клавиш клавиатуры с помощью addEventListener, сдвигающие и переворачивающие текущую игровую фигуру.

Поле представляет собой двумерный массив 10\*20, хранящий два состояния: есть блок (1), нет блока (0). Это необходимо для расчёта коллизий. Когда фигура «приземляется» происходит проверка заполненных линий, если линии найдены и удалены, объект Score накапливает очки и хранит «строку» из 4 символов в виде массива для удобства записи счёта в HTML объект.

1. Описание средств JavaScript для взаимодействия с DOM:
2. Перехватываемые события:
3. Объект Event:
4. Вывод: