Snake Game – Theory Documentation

1. Overview

The Snake game is built using simple logical systems working together. The game is not graphics-first; it is logic-first. The visuals only represent data.

2. Game Board (Grid Concept)

The board is a fixed grid (for example 20×20). Each cell can be empty, contain a snake segment, or contain food. CSS Grid or Canvas can be used to visually represent this structure.

3. Snake Data Structure

The snake is represented as an array of coordinates. The first element is the head and the last element is the tail. Movement works by adding a new head and removing the tail.

4. Direction System

Direction is stored as a value such as UP, DOWN, LEFT, or RIGHT. Opposite direction changes are restricted to avoid instant self-collision.

5. Game Loop

The game loop runs at a fixed interval. Each tick updates the snake position, checks collisions, and triggers re-rendering.

6. Food System

Food is a single coordinate placed randomly on the grid. When the snake eats food, it grows and new food is generated.

7. Collision Detection

Two collision types exist: wall collision and self collision. If either occurs, the game ends.

8. Rendering Logic (React)

React renders UI based on state. Each grid cell checks whether it matches snake or food coordinates and applies styles accordingly.

9. Input Handling

Keyboard input updates only the snake direction. Movement always happens inside the game loop.

10. Required State Variables

The game only needs snake, direction, food, score, and gameOver state. Keeping state minimal prevents bugs.

11. System Flow Summary

Keyboard updates direction → Game loop moves snake → Collision & food logic updates state → React re-renders UI.