Snake Game - Complete Game Design Document

1. GAME OVERVIEW

What is Snake? Snake is a classic arcade game where you control a growing snake on a grid. The snake moves automatically and continuously in one direction. Your goal is to eat food to grow longer and score points, while avoiding crashing into walls or your own body.

Core Objective: Get the highest score possible by eating as much food as you can before the game ends.

2. CORE MECHANICS (How the Game Works)

2.1 The Snake

What it is:

- The snake is made up of connected segments (think of it like a chain)
- Each segment occupies one square on the grid
- The snake has a "head" (front segment) and a "tail" (back segment)

Starting Condition:

- The snake begins with 3 segments
- Starts in the middle of the screen
- Initially moves in one direction (let's say RIGHT)

How it moves:

- The snake moves **continuously** - it never stops moving
- Every 150 milliseconds (0.15 seconds), the snake moves one square forward
- Movement works like this:
    1. The head moves to the next square in the current direction
    2. Each body segment moves to where the segment in front of it was
    3. The tail segment disappears (unless the snake just ate food)

2.2 Player Controls

Input Method:

- Arrow Keys: UP, DOWN, LEFT, RIGHT
- Alternative: WASD (W=up, A=left, S=down, D=right)

How controls work:

- Pressing a key changes the snake's direction
- The snake will continue moving in that direction until you press another key
- **Important rule:** You cannot reverse direction instantly

    - If moving RIGHT, you cannot go LEFT
    - If moving UP, you cannot go DOWN
    - This prevents the snake from running into itself

2.3 Food System

What is food:

- Food is a single object that appears on the grid
- Represented by a special character (like *, @, or ●)
- Only ONE piece of food exists at a time

Food Spawning:

- When the game starts, one food appears at a random empty square
- When the snake eats food, a new food immediately spawns at another random empty square
- Food **never** spawns on the snake's body or on walls

Eating Food:

- The snake "eats" food when its head moves onto the square containing food
- When food is eaten:
    1. The snake grows by 1 segment (added to the tail)
    2. Score increases by 10 points
    3. New food spawns somewhere else
    4. (Optional) Speed might increase

2.4 Growth Mechanism

How the snake grows:

- Normally: tail disappears as head moves forward (constant length)
- After eating: tail stays in place for ONE move (snake grows by 1)

Technical explanation:

Normal move (no food):
1. Add new head position
2. Remove tail position
Result: same length

After eating food:
1. Add new head position
2. DON'T remove tail position

Result: length + 1

3. GAME PARAMETERS (Settings & Numbers)

3.1 Grid/Board

Grid Size: 20 × 20

- The game board is a square grid
- 20 squares wide × 20 squares tall
- Total of 400 squares

- Grid includes borders (walls) on all sides

Why 20×20:
- Small enough to fit in a console window
- Large enough for interesting gameplay
- Adjustable in code if you want bigger/smaller

3.2 Speed/Timing

Starting Speed: 150 milliseconds per move
- This means the snake moves forward one square every 0.15 seconds
- About 6-7 moves per second
- Fast enough to be engaging, slow enough for beginners

Speed Progression (Optional Feature):
- Every 50 points (5 food items), speed increases
- Decrease delay by 10-20ms each time
- Minimum speed: ~50ms (very fast, expert level)

Example progression:
- 0-40 points: 150ms (starting speed)
- 50-90 points: 130ms (faster)
- 100-140 points: 110ms (even faster)
- 150+ points: 90ms (very challenging)

3.3 Snake Properties

Starting Length: 3 segments

- Simple enough to control at the start
- Allows immediate growth to feel progression

Starting Position:

- Center of the grid (position 10, 10)
- Facing RIGHT

Starting Direction:

- RIGHT (moving →)

3.4 Scoring

Food Value: 10 points per food

- Simple scoring system
- Easy mental math (5 foods = 50 points)

Score Display:

- Shows current score at top of screen
- Updates immediately when food is eaten
- Format: "Score: 120"

Optional - High Score:

- Track the best score of the session
- Display alongside current score
- Persists until program closes

4. WIN/LOSS CONDITIONS

4.1 How the Game Ends (Loss Conditions)

There are TWO ways to lose:

1. Wall Collision

- The snake's head touches any wall/border
- Walls are the # characters around the edge
- Example:

```
####
# ■→ # ← Snake head hits wall
####

GAME OVER
```

2. Self Collision

- The snake's head touches any part of its own body
- This becomes more likely as the snake grows longer
- Example:

```
■■■■
■ ■
↓ ■ ← Snake head hits own body

GAME OVER
```

4.2 Win Condition

There is NO traditional "win"

- The game is endless - you play until you crash
- Goal is to achieve the highest score possible
- Theoretical maximum score: filling the entire grid (very difficult!)

Why no win condition:

- Encourages replayability
- Players compete against themselves or friends for high scores
- Classic arcade-style gameplay

Theoretical Perfect Game:

- Snake fills every square on the 20×20 grid
- That's 400 segments long
- Score would be: (400 - 3) × 10 = 3,970 points
- Practically impossible due to increasing speed and difficulty

---

5. GAME FLOW (How a Game Session Works)

Step-by-Step Gameplay:

1. Game Start

- Display title screen / instructions
- Wait for player to press START (e.g., ENTER key)

2. Initialize Game

- Create 20×20 grid with walls
- Place snake at center (3 segments, facing right)
- Spawn first food at random location
- Set score to 0

3. Main Game Loop (repeats continuously)

- Wait 150ms (or current speed)
- Check for keyboard input (direction change)
- Move snake one square forward
- Check if snake head is on food → eat it, grow, spawn new food
- Check if snake hit wall or itself → game over
- Redraw the screen with updated positions
- Repeat

4. Game Over

- Stop the game loop
- Display "GAME OVER" message
- Show final score
- Ask: "Play again? (Y/N)"

5. Restart or Exit

- If YES: go back to step 2

- If NO: close program

6. OPTIONAL FEATURES (To Add Later)

6.1 Difficulty Levels

- **Easy:** Slower speed (200ms), no speed increase
- **Medium:** Normal speed (150ms), gradual increase
- **Hard:** Fast speed (100ms), rapid increase

6.2 Power-ups

- **Slow Down:** Temporary speed reduction
- **Invincibility:** Can pass through walls/self for 3 seconds
- **Double Points:** Next food worth 20 points

6.3 Obstacles

- Add random wall pieces inside the grid
- Makes navigation more challenging

6.4 Different Game Modes

- **Classic:** Standard rules
- **No Walls:** Snake wraps around screen edges
- **Timed:** Survive for 60 seconds

6.5 Enhanced UI

- **Pause Function:** Press P to pause
- **Lives System:** 3 lives before full game over
- **Visual Improvements:** Colors, animations

---

7. TECHNICAL SPECIFICATIONS (For Programming)

7.1 Data Structures Needed

Snake Body:

- Use std::deque<Point> or std::vector<Point>
- Each Point stores x,y coordinates
- Example: {(10,10), (9,10), (8,10)} for 3-segment snake

Grid Representation:
2D array: char grid[20][20]
Characters represent different things:
' ' = empty space
'#' = wall
'O' = snake head
'o' = snake body
'*' = food

Direction:

- Use enum: enum Direction {UP, DOWN, LEFT, RIGHT}
- Or store as dx, dy values

7.2 Key Functions Needed

- initGame() - Set up initial state
- drawBoard() - Render grid to console
- getUserInput() - Check for key presses
- moveSnake() - Update snake position
- checkCollision() - Detect wall/self hits
- spawnFood() - Generate random food position
- updateScore() - Increase score

7.3 Game Loop Pseudocode

```
while (gameRunning):
  clearScreen()
  drawBoard()
  displayScore()

  if (keyPressed):
    updateDirection()

  sleep(currentSpeed)
  moveSnake()

  if (snakeHeadOnFood):
    growSnake()
    spawnNewFood()
    increaseScore()

  if (collision):
    gameRunning = false
    showGameOver()
```

8. USER EXPERIENCE DETAILS

8.1 Controls Display

Show controls on screen during gameplay:
Controls: Arrow Keys or WASD | P: Pause | Q: Quit

8.2 Visual Feedback

- Screen clears and redraws each frame
- Score updates immediately
- Clear GAME OVER message when losing

8.3 Instructions Screen

Before the game starts, show:

=== SNAKE GAME ===

GOAL: Eat food (*) to grow and score points!

CONTROLS:
  Arrow Keys or WASD to change direction

RULES:
  - Don't hit the walls (#)
  - Don't hit yourself
  - Eat food to grow

Press ENTER to start...