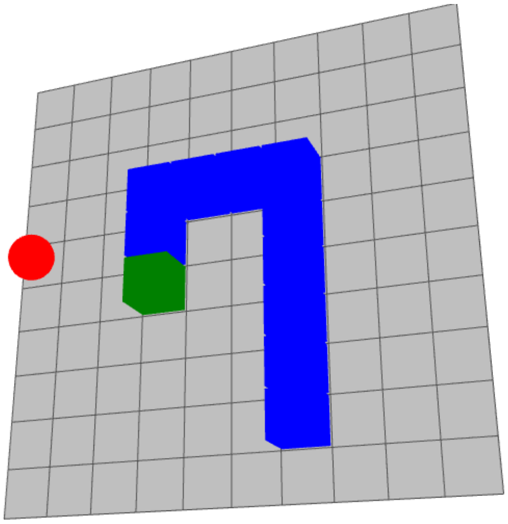
Introduction to Computer Graphics
K. Jünemann

Assignment 1: A simple snake game

The idea is to write a very simple snake game.



1 Requirements

1. Create a gray playing field in the x-y-plane of size 10×10 with the origin of world space located at the center of the playing field.

2. Divide this playing field into units cells of size 1×1 and visualize them with a grid.

   Hint: use `THREE.GridHelper`

3. Add a snake, which initially consists of a single cube at rest, at a random unit square of the playing field.

4. Place a red ball of unit diameter at a random cell in the playing field. When the head of the snake hits the red ball grow the snake by one unit and reposition the ball to a random unit cell which is not covered by the snake.

5. Move the snake forward by one unit every 250ms.

   Hint: use `setInterval`

6. Use the arrow keys to specify the direction into which the snake moves. Make sure that the snake is at rest initially and starts moving only at the first key stroke.

7. Use a green cube for the head of the snake and blue cubes for the rest of the snake. The edge length of each cube should be 95% of the side length of a unit cell so that a small gap is visible between the cubes of the snake.

8. Use the alert function to report game over either when the head of the snake moves beyond the boundaries of the playing field or when the snake intersects itself. Also report the length of the snake when the game is over.

Optional: Use the three.js audio interface (see documentation) to play a sound whenever the snake eats a ball and when the game is over. This only works if you run your application using a web server.

2 Hints

You may implement the snake as a double ended queue (deque). To support this a deque implementation from the *LearnersBucket* platform[1] has been copied to the end of the assignment code template. If you do not want to use this code feel free to delete it from the final version of your assignment.

3 Coding style

Stick to the coding style guide which can be found in the README.md file for chapter 3 in the gitlab repository.

4 Handing in the solution

No group work allowed. Every course partiticipant has to write her or his *own* code!
Implement your entire solution within the file SnakeGame.js. Feel free to change this file as you like but do not add any further files.
Submit your code by creating a zip-archive from the folder Assignment1 and upload it on the Moodle page.

[1] https://learnersbucket.com/tutorials/data-structures/implement-deque-data-structure-in-javascript/