

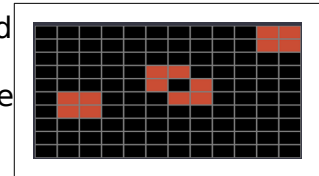
Game of Life: Demo Checklist

This document describes how we will test and grade the live demo of **Game of Life** project. The demo will be performed on the provided VSCode extension.

1 Game Reset State

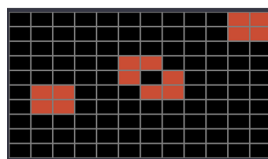
Launch the extension with the **last submitted project file**. The game should be initialized as follows:

1. [1 point] LEDs: the pattern on the right should be displayed.
2. [1 point] 7-segment displays show X001, where X stands for an undefined value.

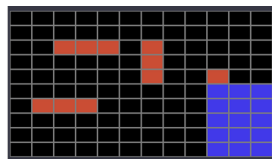


2 INIT State

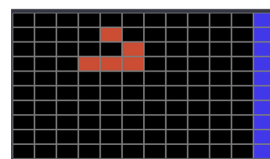
1. [1 point] Buttons b0, b1, and b2: Pressing these buttons should increment, in steps of one, the value shown on the 7-segment display. Button 0 configures the unit, button 1 the tens and button 2 the hundreds. The digits should be displayed in hexadecimal format: 0-1-2-3-4-5-6-7-8-9-a-b-c-d-e-f.
2. [2 points] Button jc: Pushing button jc should result in the display showing the next predefined seed, in the following sequence (from left to right). After going through all predefined seeds, the game should transition to the **random** state.



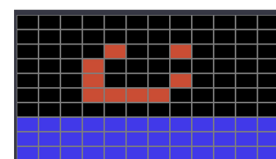
seed 0



seed 1



seed 2



seed 3

3 RAND State

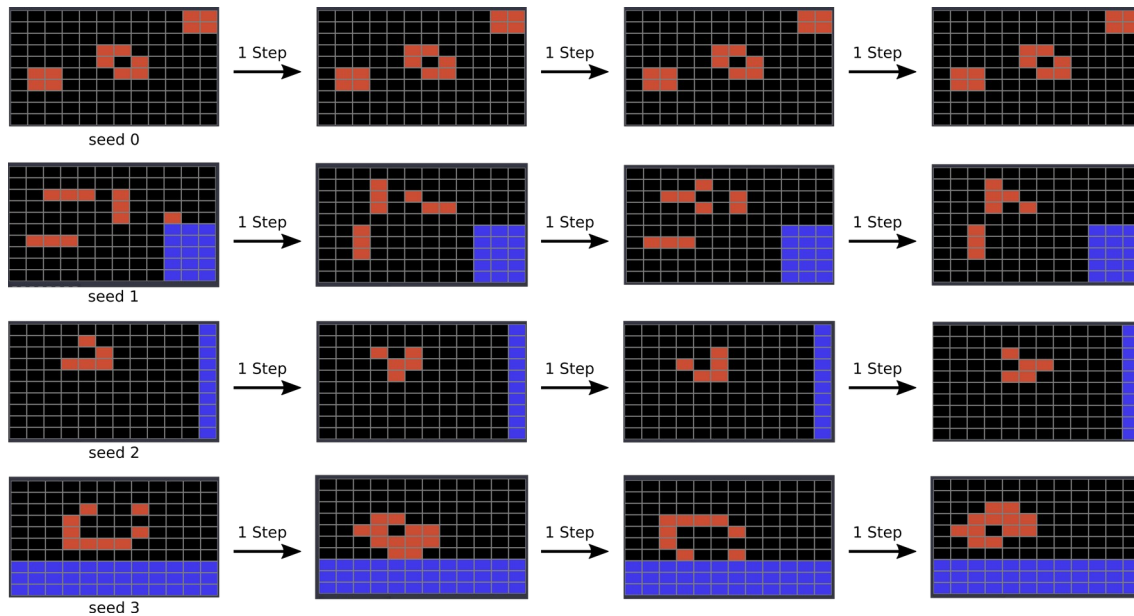
1. [2 points] Button jc: Continuing to press button jc should generate a new random state every time the button is pressed.
2. [1 point] Buttons b0, b1, and b2: Pressing these buttons should have the same effect as described in **INIT** state.
3. [2 points] Button jr: Pressing this button starts the game.

4 RUN State

1. [2 points] Button jc: start/pause button. When paused, the LEDs should not update and the 7-seg should not decrement.
2. [2 points] Button jr: increases the speed of the game.
3. [2 points] Button jl: Decreases the speed of the game. Verify that the game slows down but never stops before the number of steps drops to zero.
4. [2 points] Button jt: Wait for the game to reach a configuration where nothing changes anymore, then press button jt. This should replace the current game state with a new random one.
5. [2 points] Button jb: Reset button. Pressing it should bring you back to the initial state explained in **Game Reset State**.

5 GoL Algorithm

Finally, we will test that the Game of Life algorithm is correctly implemented by choosing one of the predefined seeds and running the game for a number of steps. Only if all the displayed game steps are correct, and in correct sequence, will this test be considered as **successfully passed**.



6 Grading

- Maximum score is 20 points.
- The score is computed as follows:

$$Score = \begin{cases} \sum \text{points, if GoL algorithm test passes} \\ \frac{1}{2} \sum \text{points, if GoL algorithm test does not pass} \end{cases}$$

Here, the *sum of points* takes into account all the tests in Sections 1, 2, 3, and 4. GoL algorithm tests are described in Section 5.