IS2545 Deliverable 4’s Manual Testing

Yichao Chen

**WHT MANUAL TESTING?**

When I try to test the runCountinous() method in MainPanel class, I found out that there was no direct output in this method --- the outcome would displayed in the player’s panel. According to this feature, I decide to manual test this method by observing the cell’s position and status’s change.

**TEST CASE 1:**

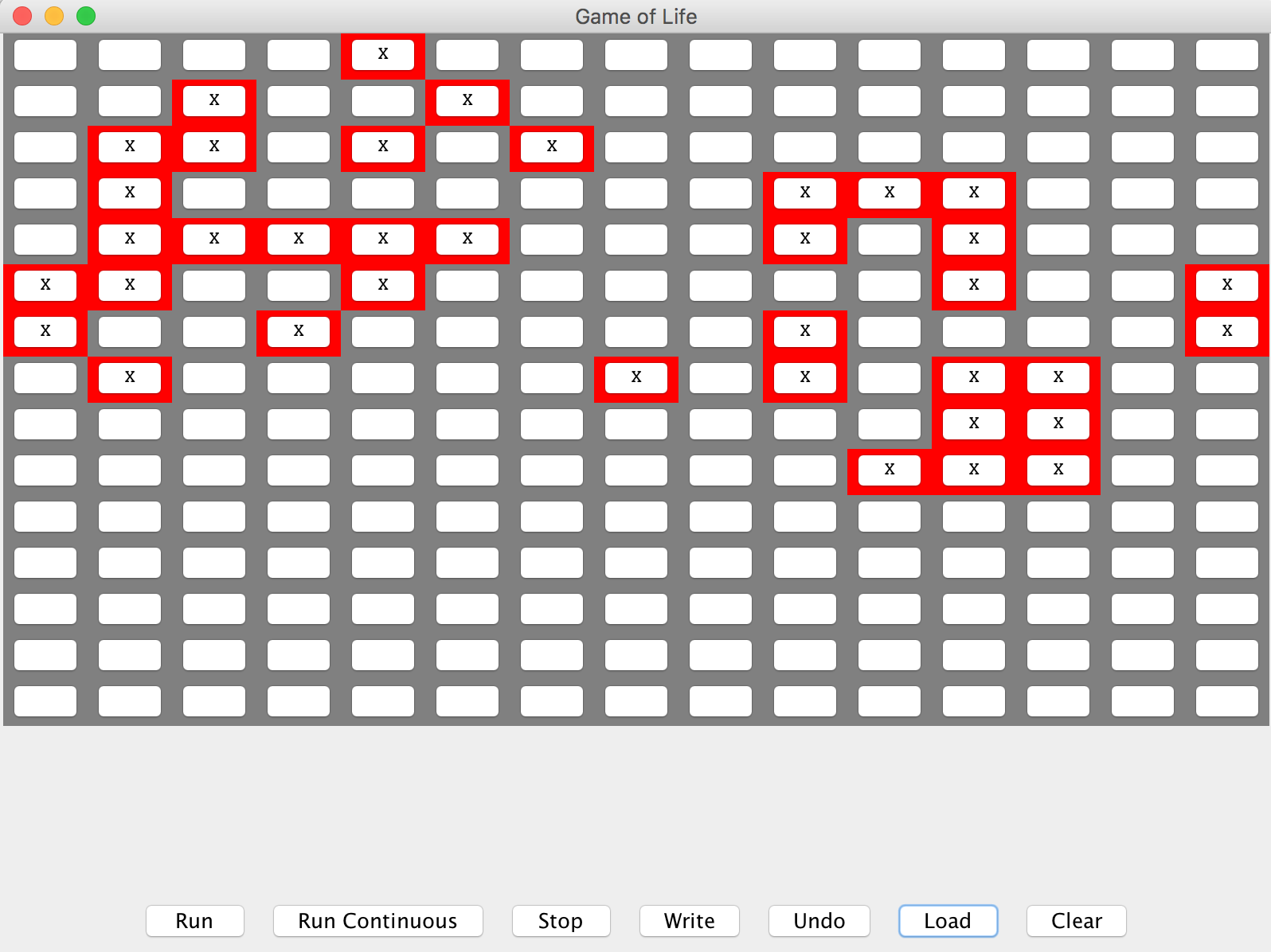
Test whether the modified method runCountinous() gives the same result in 15 \* 15 as the unmodified one gives.

**PRECONDITIONS:**

Creating a 15 \* 15 world to start the game.

**EXECUTION STEPS:**

1. Select the cells in the panel like below:



2. Click on Run Continuous button in the window.

3. Let the game runs until it reach a status that all block of cells unchanged (stay in a relative static situation.)

4. Screen shot the result:

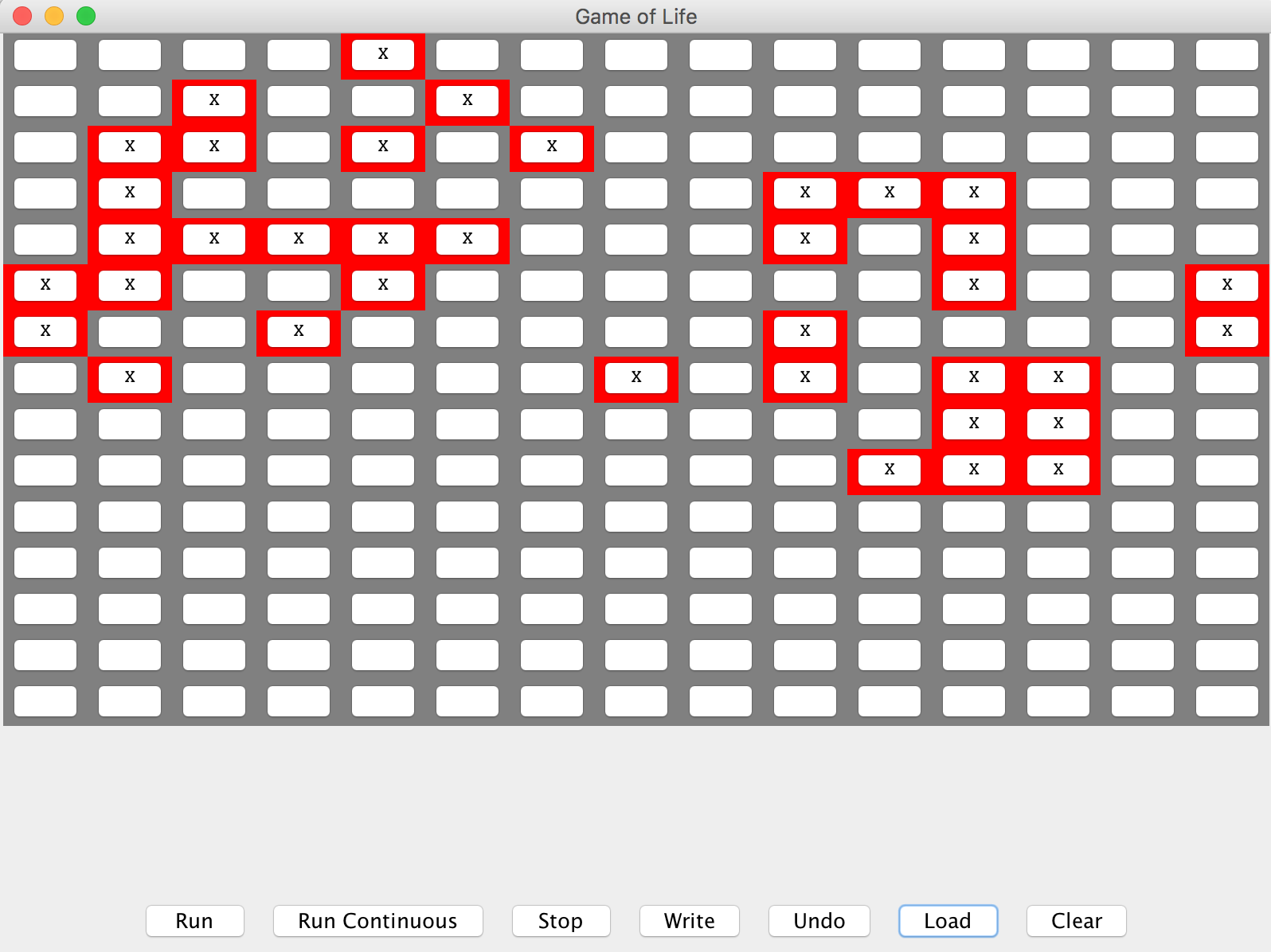


5. Exit Game of Life.

6. Implement the modified runContinous() method.

7. Start the game by the same 15 \* 15 world

8. Click the exact cells as before:



9. Click on Run Continuous button in the window.

10. Let the game runs until it reach the “frozen” status as what we do at step 3.

11. Record the cell’s position and status:



**POSTCONDITIONS:**

The modified runContinous() method’s outcome cells’ position and status are exactly the same as the original runContinuous() method’s outcome.

**TEST CASE 2 (Edge case 1):**

Test whether the modified method runCountinous() gives the same result in 15 \* 15 as the unmodified one gives.

**PRECONDITIONS:**

Creating a 15 \* 15 world to start the game.

**EXECUTION STEPS:**

1. Select all cells in the panel:



2. Click on Run Continuous button in the window.

3. Let the game runs until it reach a status that all block of cells unchanged (stay in a relative static situation.)

4. Screen shot the result:



5. Exit Game of Life.

6. Implement the modified runContinous() method.

7. Start the game by the same 15 \* 15 world

8. Click the exact cells as before:



9. Click on Run Continuous button in the window.

10. Let the game runs until it reach the “frozen” status as what we do at step 3.

11. Record the cell’s position and status:



**POSTCONDITIONS:**

The modified runContinous() method’s outcome cells’ position and status are exactly the same as the original runContinuous() method’s outcome.

**TEST CASE 3(Edge case 2):**

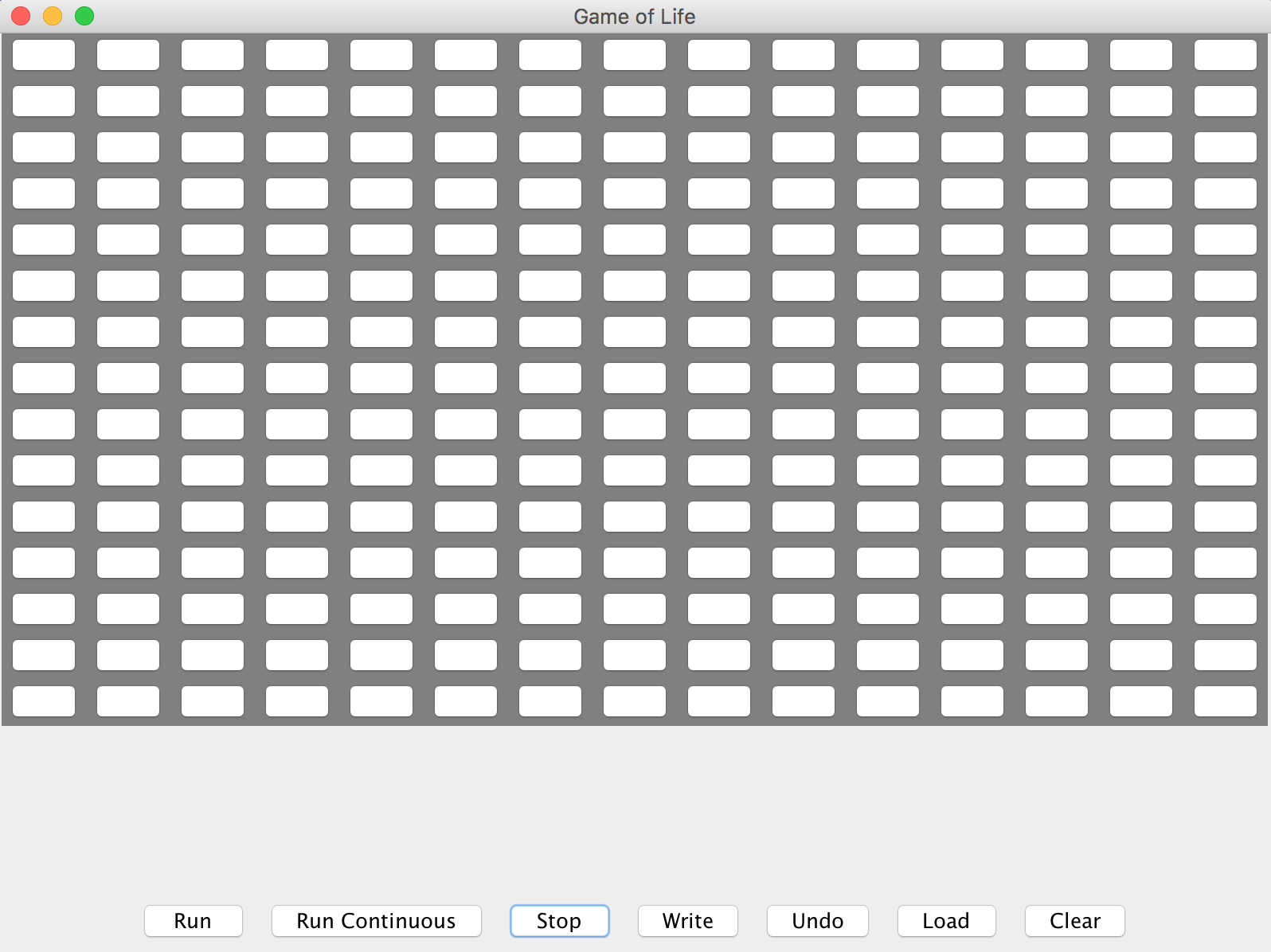
Test whether the modified method runCountinous() gives the same result in 15 \* 15 as the unmodified one gives.

**PRECONDITIONS:**

Creating a 15 \* 15 world to start the game.

**EXECUTION STEPS:**

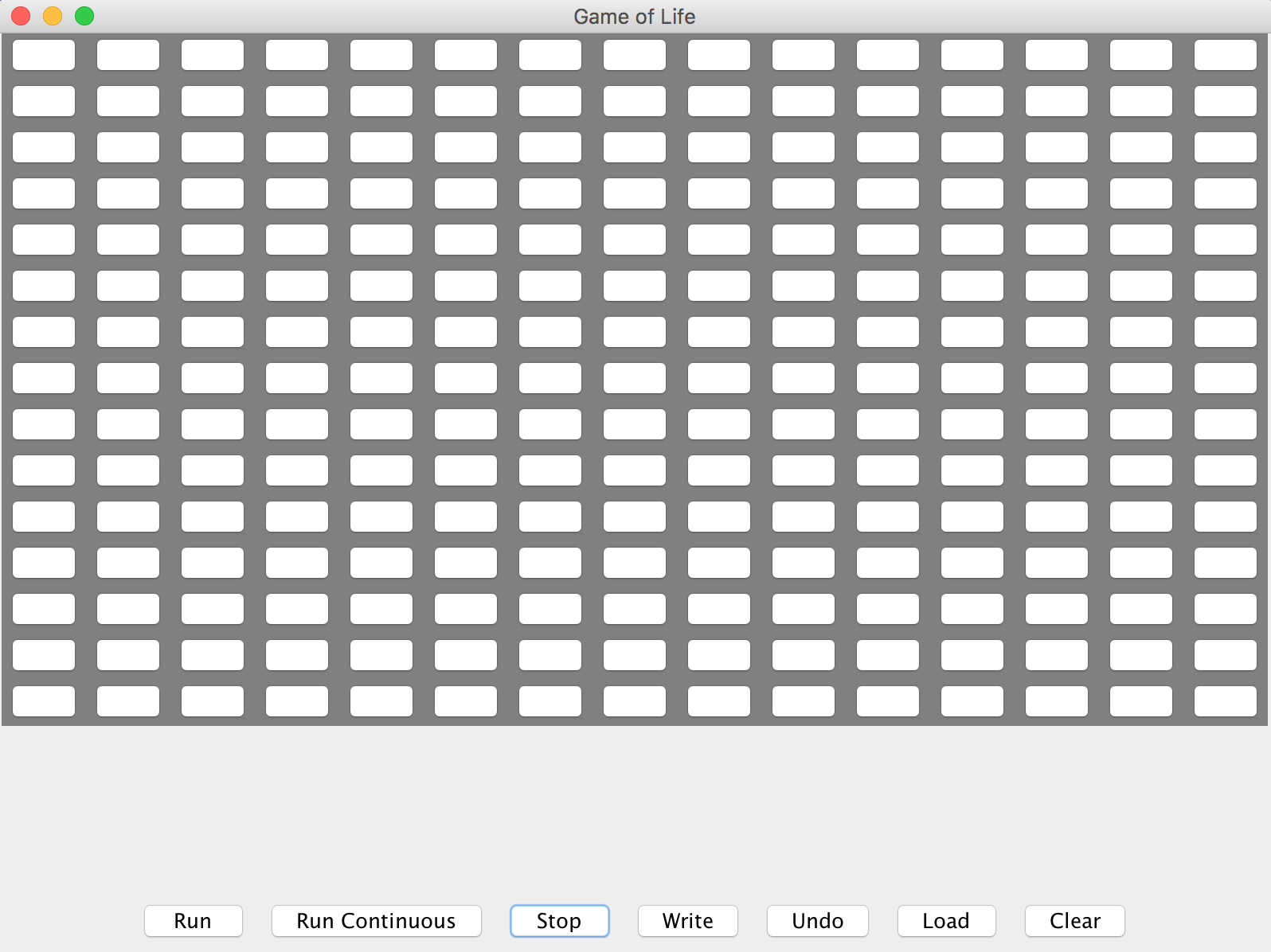
1. Don’t select any cell in the panel:



2. Click on Run Continuous button in the window.

3. Let the game runs until it reach a status that all block of cells unchanged (stay in a relative static situation.)

4. Screen shot the result like below:

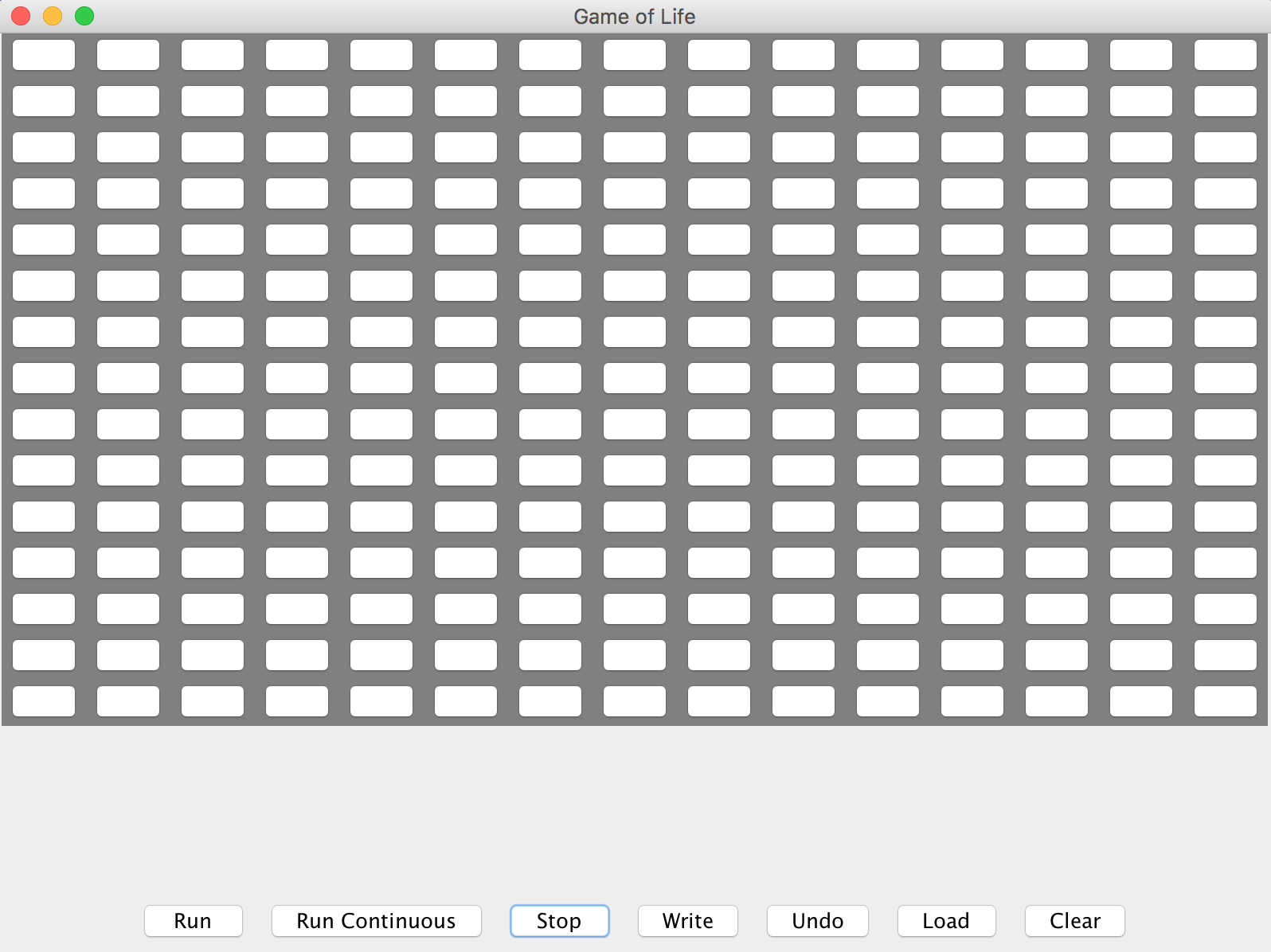


5. Exit Game of Life.

6. Implement the modified runContinous() method.

7. Start the game by the same 15 \* 15 world

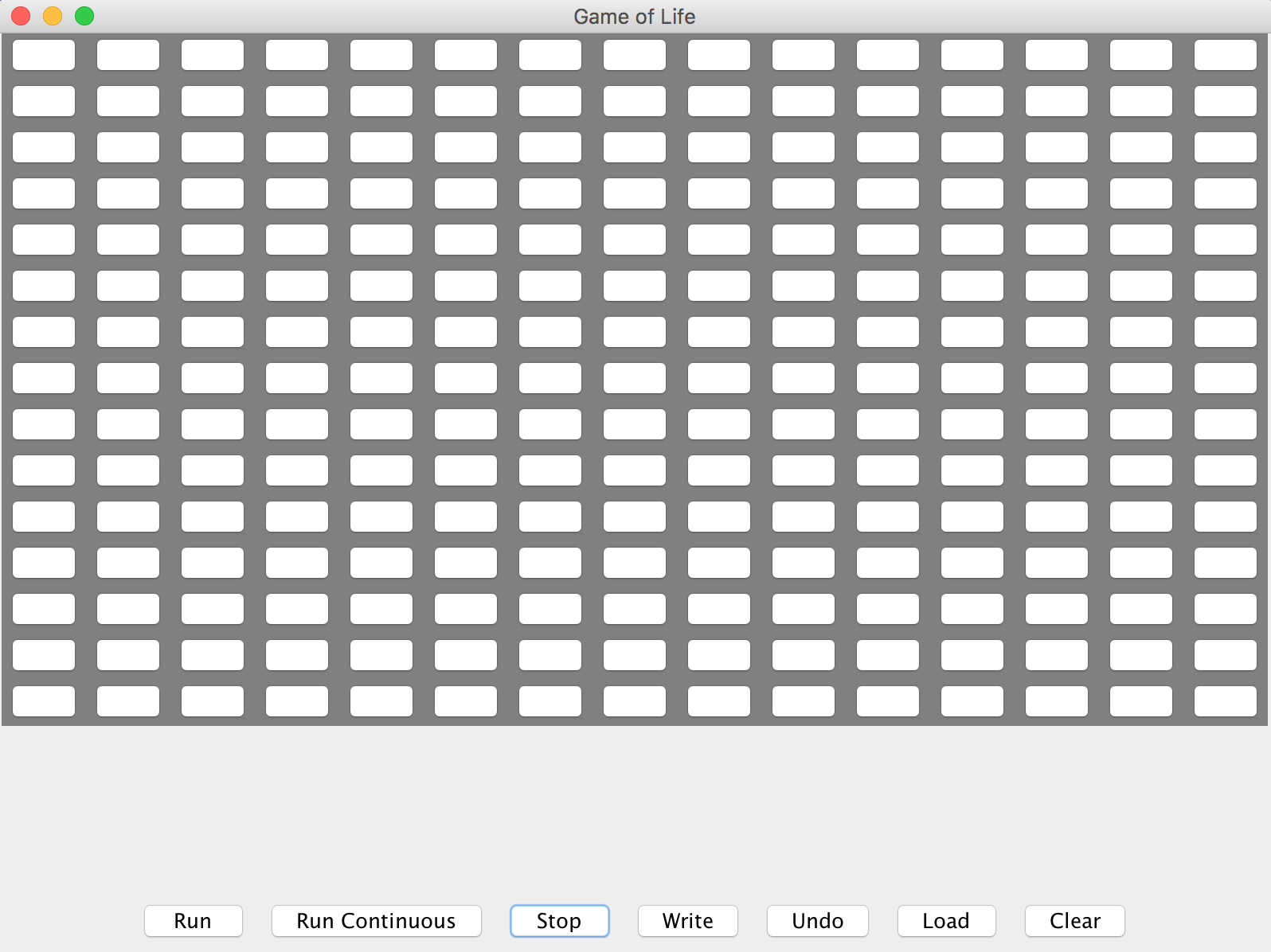
8. Click the exact cells as before:



9. Click on Run Continuous button in the window.

10. Let the game runs until it reach the “frozen” status as what we do at step 3.

11. Record the cell’s position and status:



**POSTCONDITIONS:**

The modified runContinous() method’s outcome cells’ position and status are exactly the same as the original runContinuous() method’s outcome.