| | |
|---|---|
| Name: | Hasan Yahya |
| Class: | BSE-7A |
| Roll No: | 22L-7971 |
| Subject: | Simulation and Modeling |

## Game of Life:

Game of Life is a simple 2D CA that is based on the following rules:

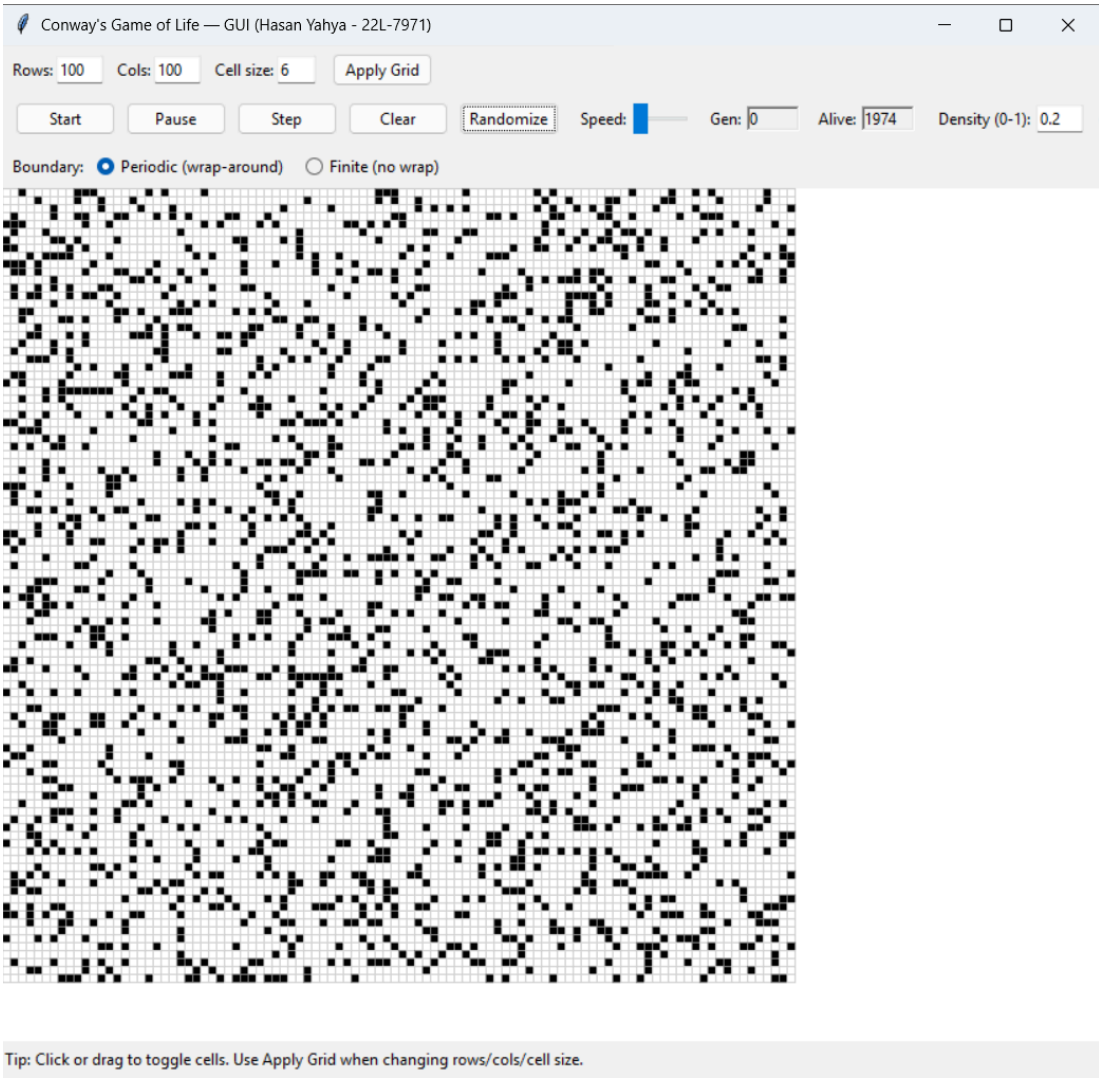
- A filled Cell is alive and an empty/white cell is dead.
- A dead (quiescent) cell will turn into a living (active) cell if and only if it is surrounded by exactly three living cells.
- A living cell will remain alive if and only if it is surrounded by two or three other living cells. Otherwise it will die.

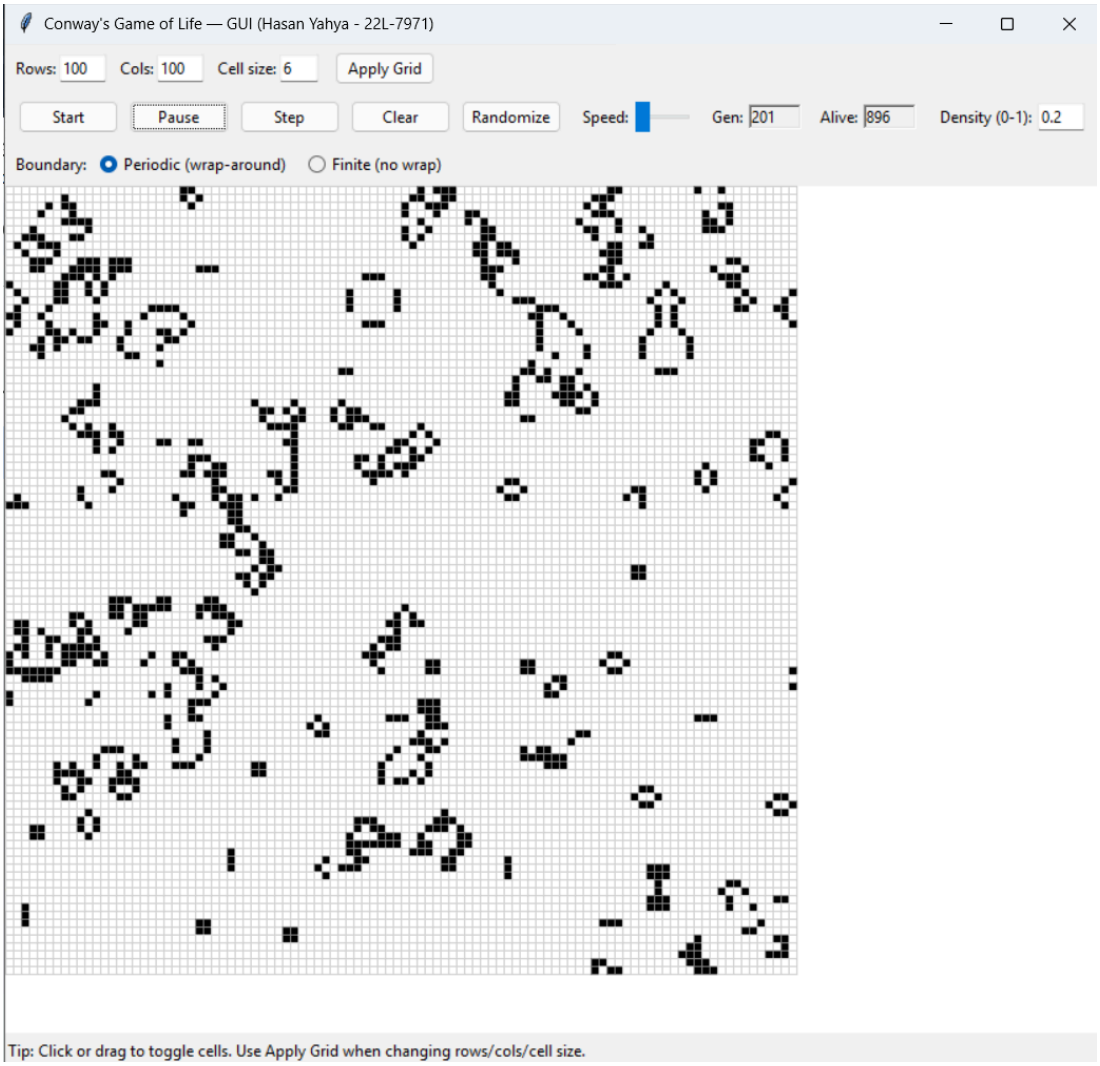
In the following report i have used a Custom Program generated by Chatgpt and fixed by me to Simulate the Game of Life and see how it works in different densities and patterns.

## Density based Simulation - Macro View (Test 1):

For the density based Simulation, I am using a Randomizing function that gives random Inputs for different densities (ie 0.2 density means 20 percent cells are alive with a leeway of +5 or -5 percent that is random). The densities chosen are (0.2, 0.5, 0.8 and 1) and the grid size will be 100 x 100. Moreover, we will have 2 modes binary and periodic giving us 8 run results. Max steps that will be allowed are 200 (or may stop earlier if all cells are killed before 200 steps are complete). ***The first image will be on initial state of grid and 2nd image will be of the state after the 200 steps have been completed.***

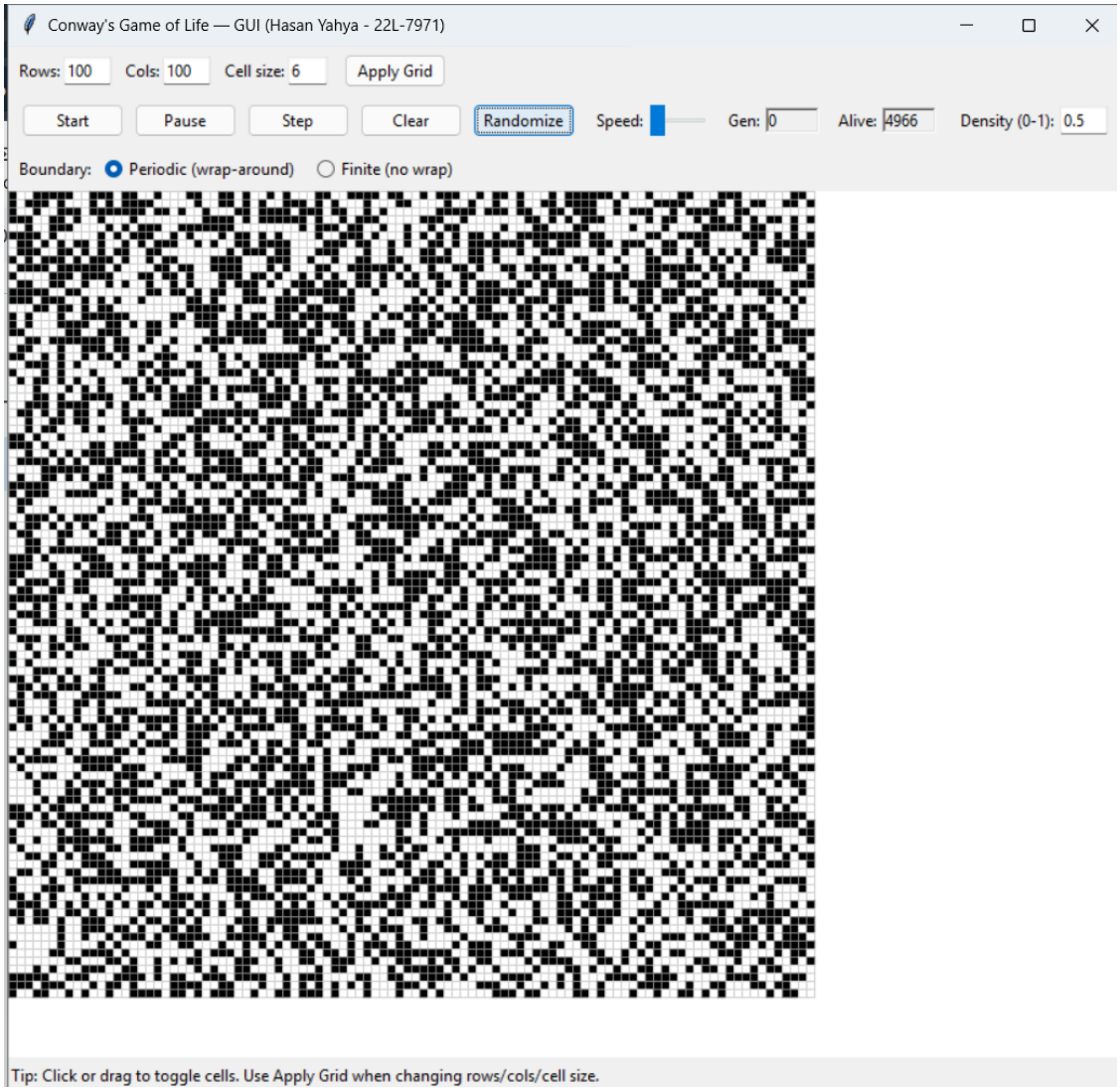
### 1. Periodic (Wrap Around with 100 x 100 Grid) & 0.2 Density:

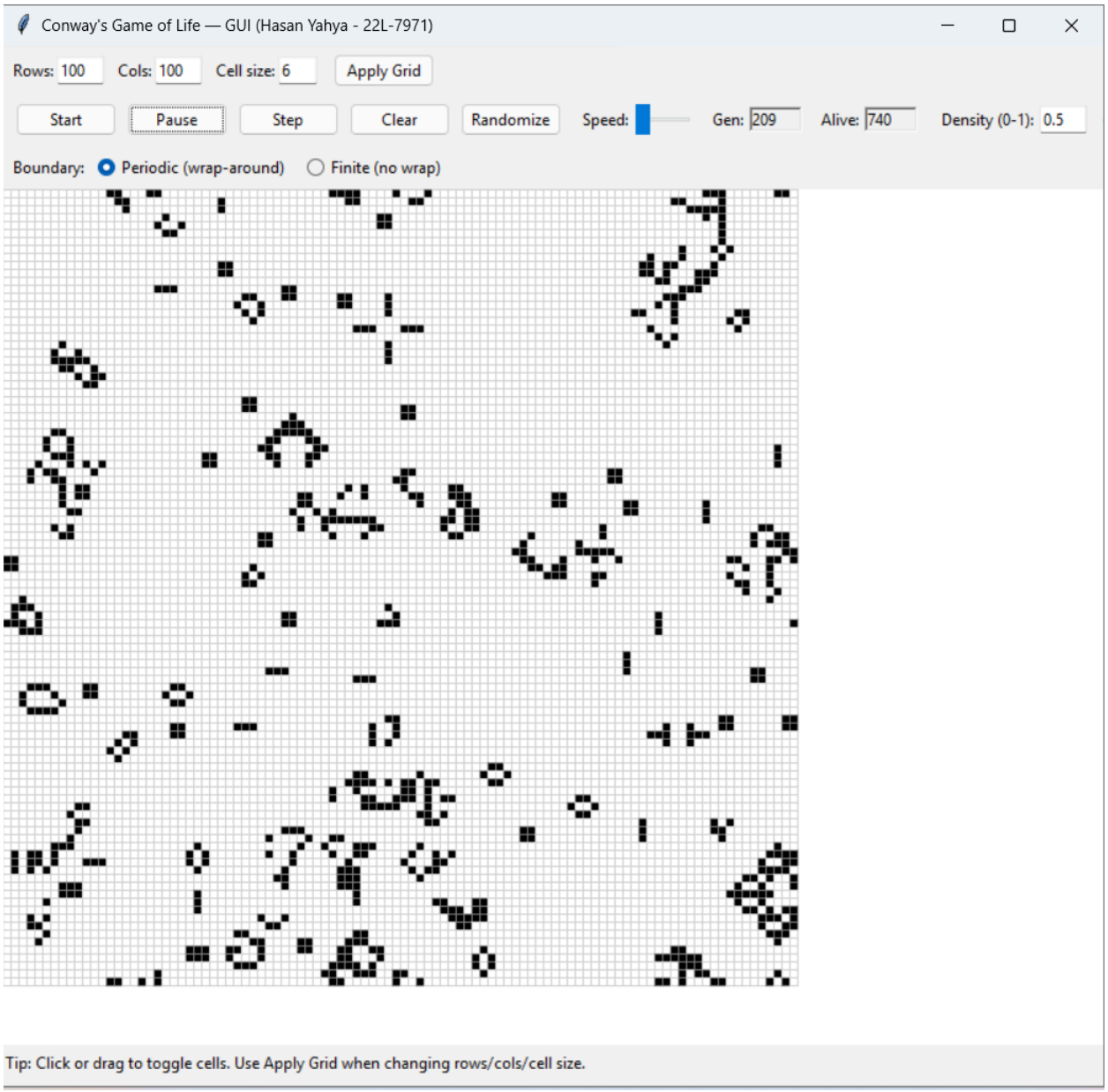

Conway's Game of Life — GUI (Hasan Yahya - 22L-7971)
Rows: 100
Cols: 100
Cell size: 6
Apply Grid
Start
Pause
Step
Clear
Randomize
Speed:
Gen: 0
Alive: 1974
Density (0-1): 0.2
Boundary:
Periodic (wrap-around)
Finite (no wrap)
Tip: Click or drag to toggle cells. Use Apply Grid when changing rows/cols/cell size.




We started with about 1974 random alive cells in a grid of 1000 cells (100 x 100) ie around 20 percent alive. In the first 80 steps, the number got reduced till 600 alive cells (most were killed off) but after that it started increasing and the population stayed consistent within the range of 800 to 900 alive cells from 80th step till 200th step (neither increasing too much not decreasing).

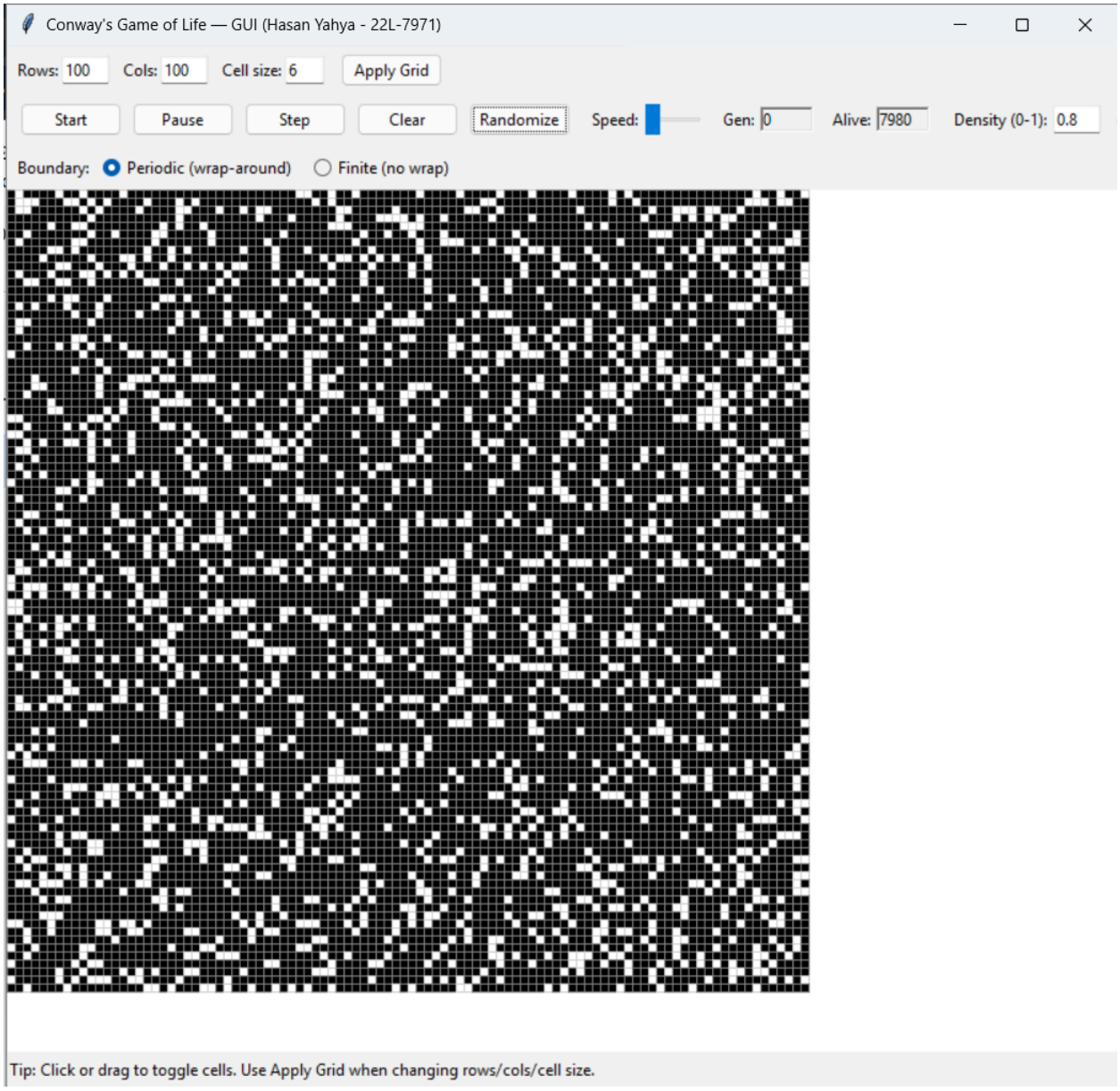
## 2. Periodic (Wrap Around with 100 x 100 Grid) & 0.5 Density:

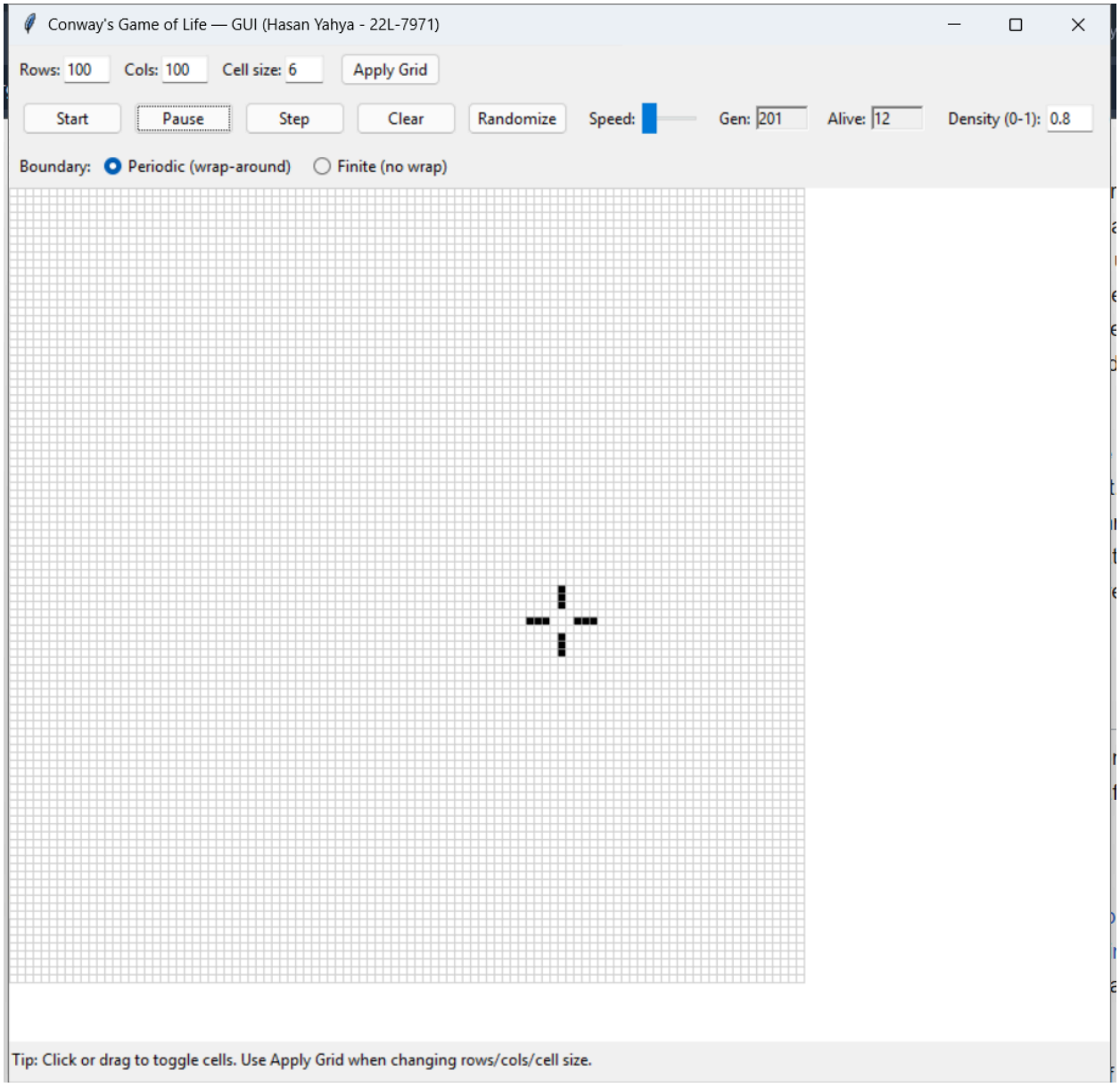

Conway's Game of Life — GUI (Hasan Yahya - 22L-7971)
Rows: 100
Cols: 100
Cell size: 6
Apply Grid
Start
Pause
Step
Clear
Randomize
Speed:
Gen: 0
Alive: 4966
Density (0-1): 0.5
Boundary:
Periodic (wrap-around)
Finite (no wrap)
Tip: Click or drag to toggle cells. Use Apply Grid when changing rows/cols/cell size.



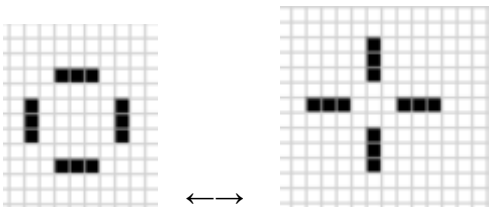

Started with 4966 alive cells, Within the first 10 steps the population of alive cells got halved to 2000. By the 50th step it remained in the 1500 to 2000 range but by the 120th step it was reduced to 600 to 900 alive cell range. So, first it was reduced exponentially then it stayed consistent until being reduced again and at the end remained in the 600 to 900 range (never completely died off).

### 3. Periodic (Wrap Around with 100 x 100 Grid) & 0.8 Density:

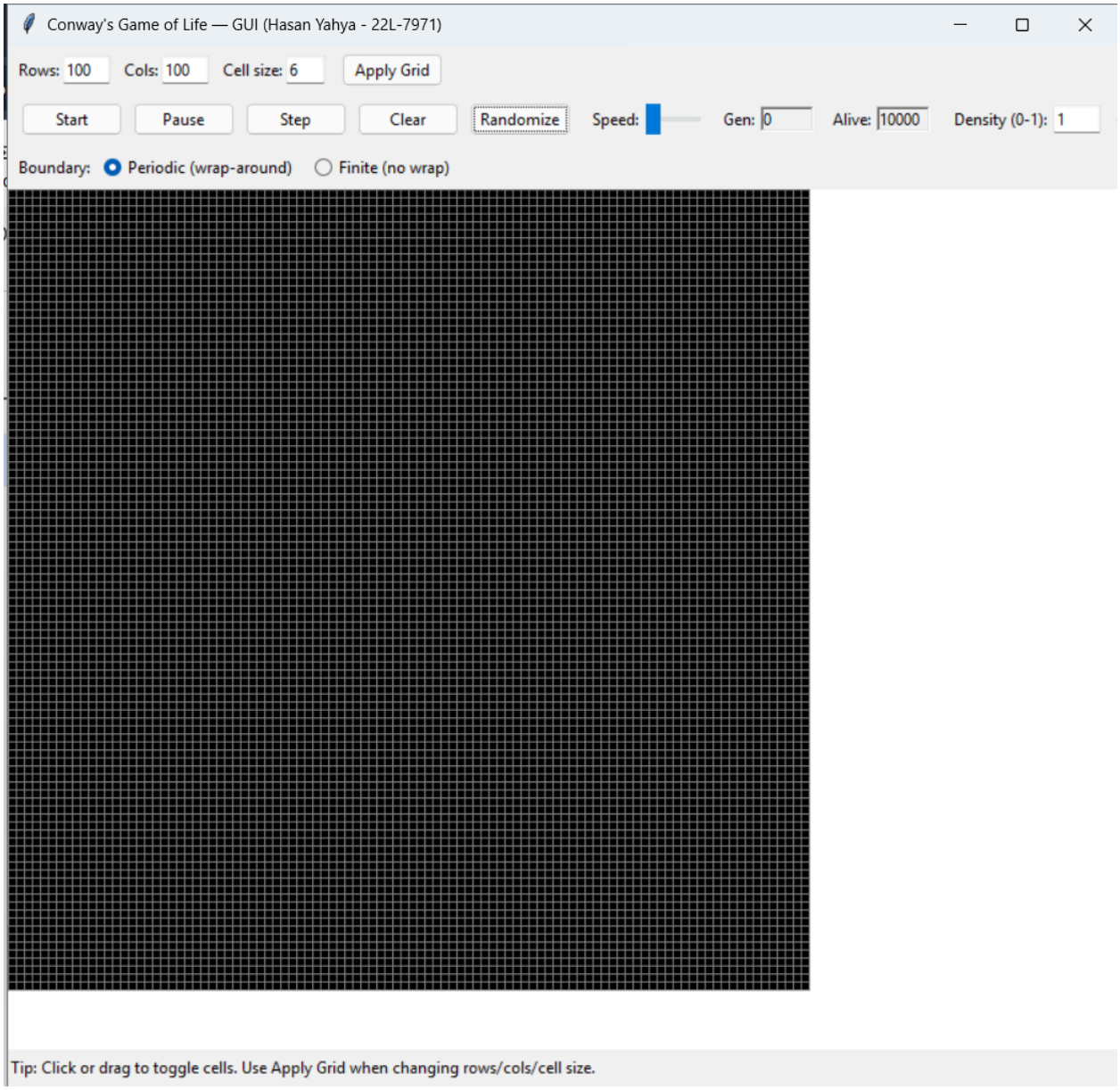

Conway's Game of Life — GUI (Hasan Yahya - 22L-7971)
Rows: 100
Cols: 100
Cell size: 6
Apply Grid
Start
Pause
Step
Clear
Randomize
Speed:
Gen: 0
Alive: 7980
Density (0-1): 0.8
Boundary:
Periodic (wrap-around)
Finite (no wrap)
Tip: Click or drag to toggle cells. Use Apply Grid when changing rows/cols/cell size.

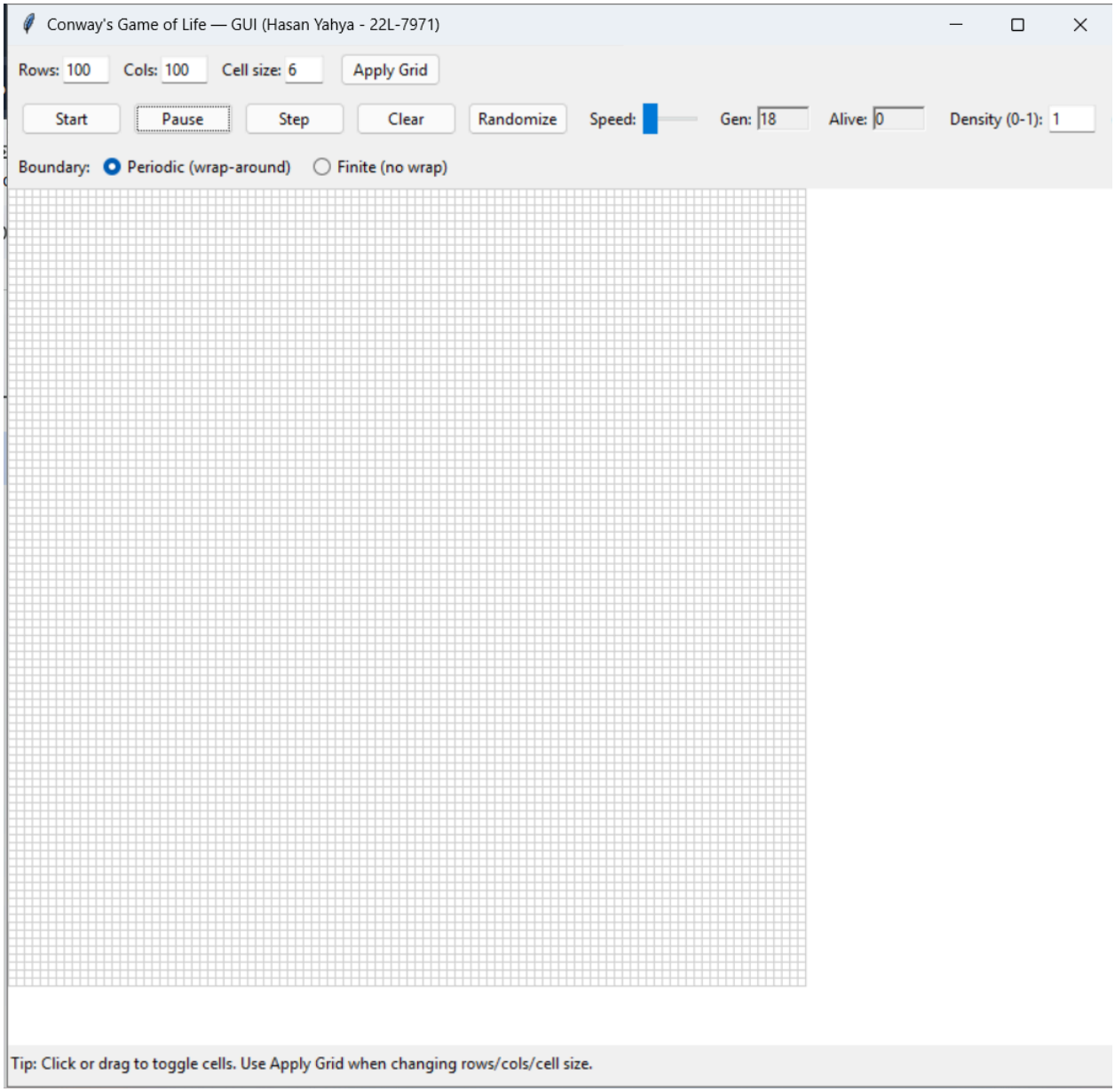



Started with 7980 cells and got reduced to 12 alive cells by the 8th step. After which two patterns for 12 alive cells each kept repeating and switching between themselves (from 8th to 200th step exactly same pattern kept repeating). The two repeating patterns are as below:



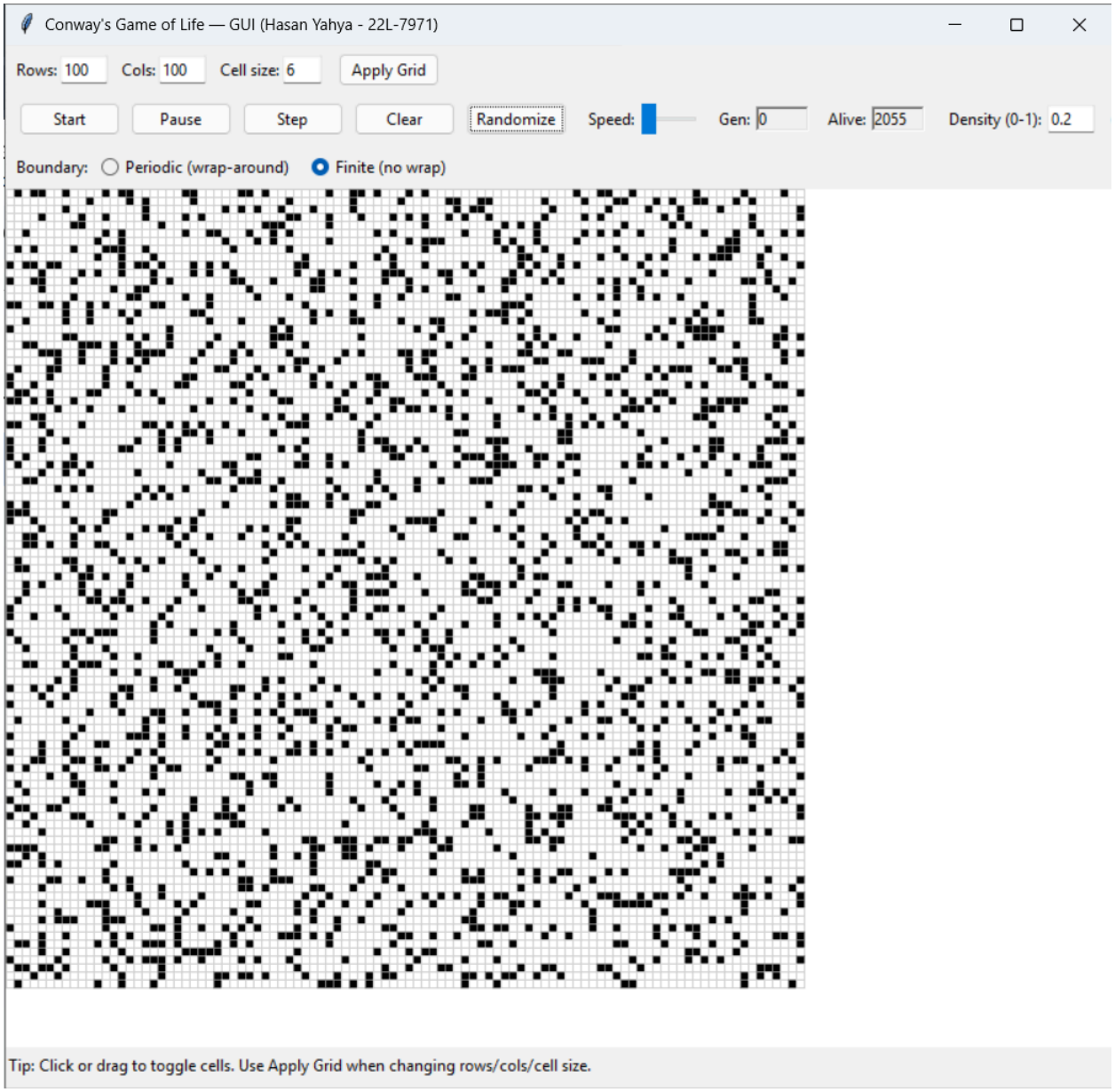
## 4. Periodic (Wrap Around with 100 x 100 Grid) & 1.0 Density:

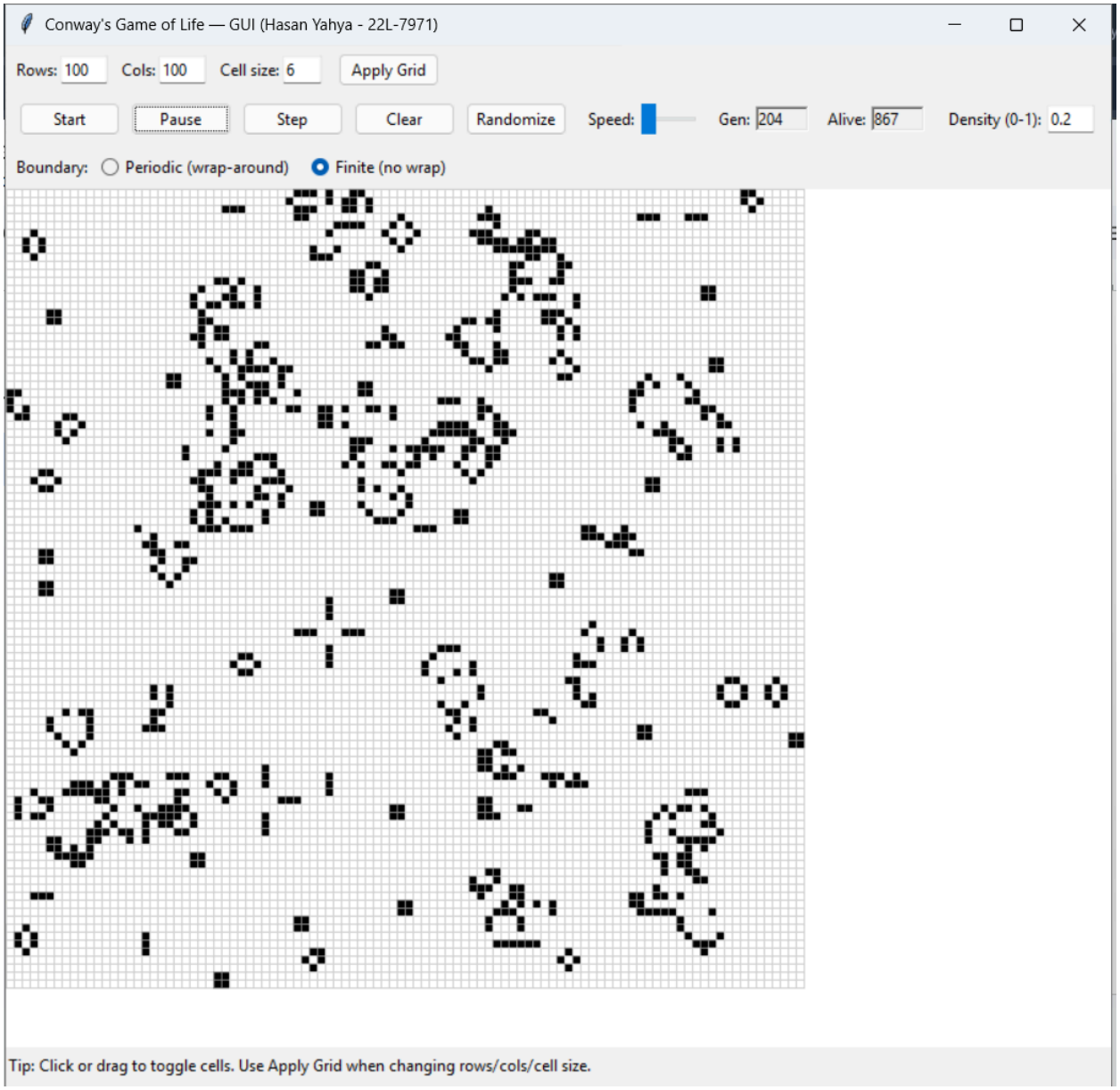

Conway's Game of Life — GUI (Hasan Yahya - 22L-7971)
Rows: 100
Cols: 100
Cell size: 6
Apply Grid
Start
Pause
Step
Clear
Randomize
Speed:
Gen: 0
Alive: 10000
Density (0-1): 1
Boundary:
Periodic (wrap-around)
Finite (no wrap)
Tip: Click or drag to toggle cells. Use Apply Grid when changing rows/cols/cell size.




Started with 10000 alive cells and by the 2nd step all were dead till the 200th step. All died.

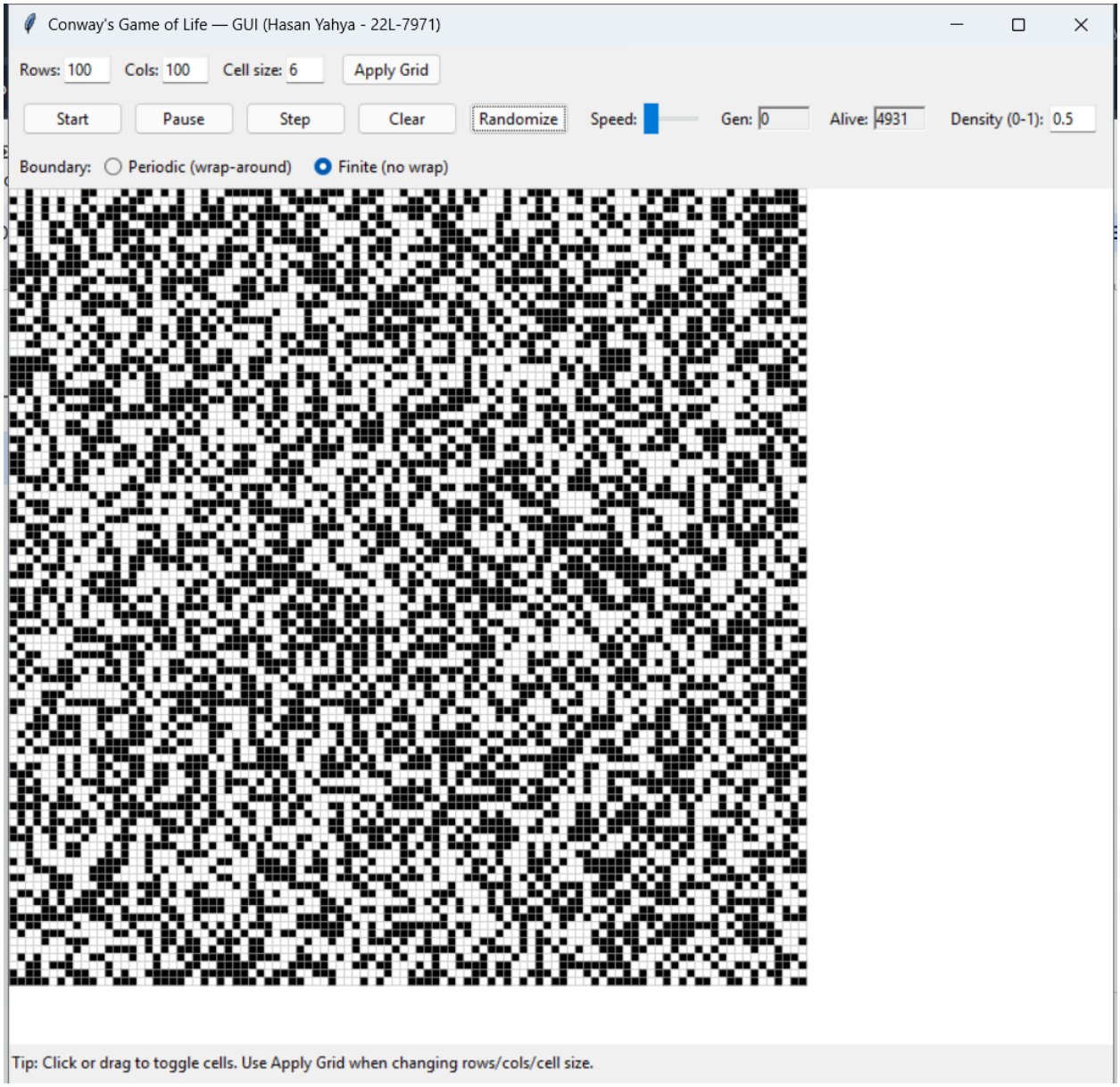
### 5. Finite (No Wrap Around with 100 x 100 Grid) & 0.2 Density:

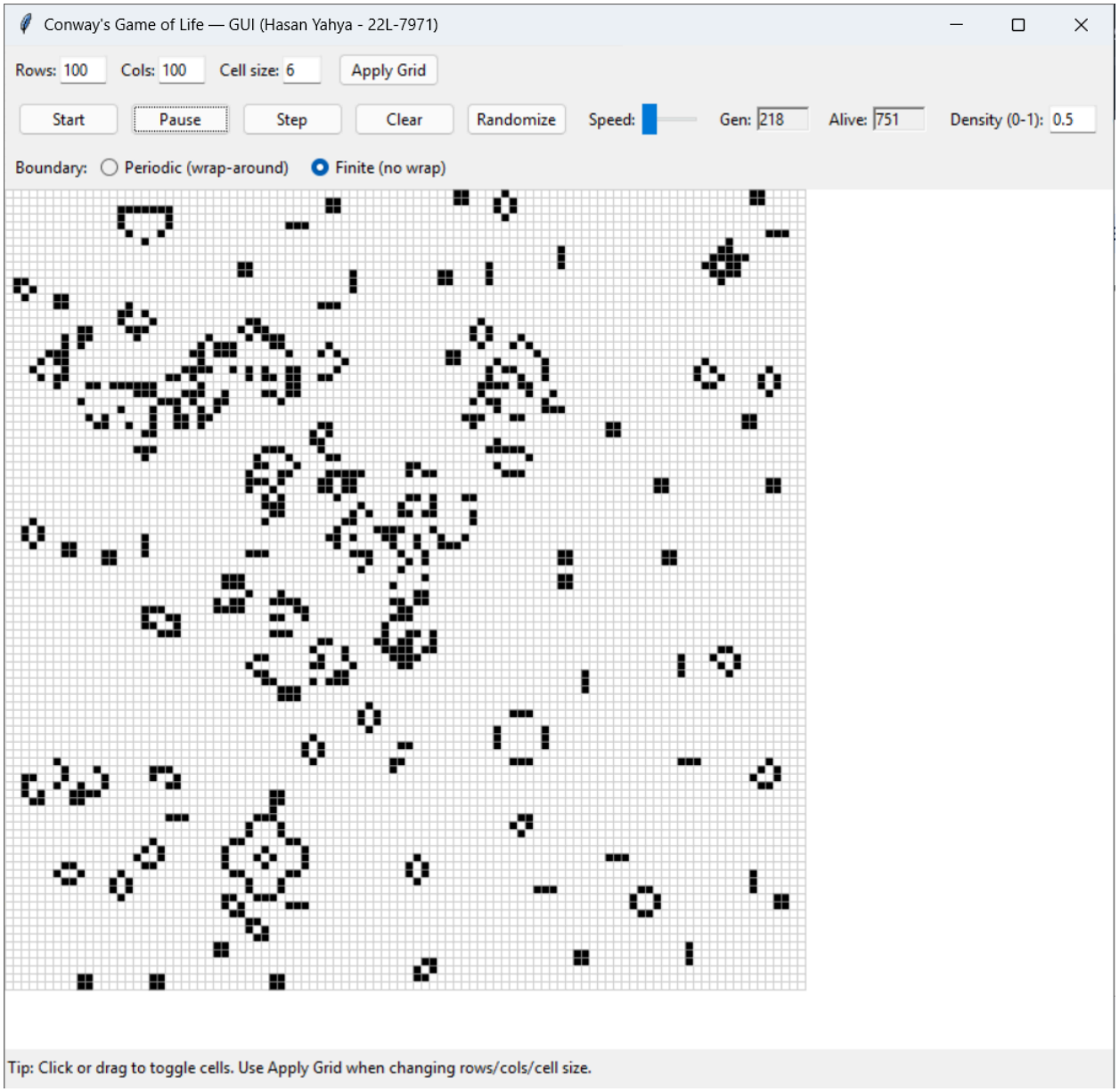

Conway's Game of Life — GUI (Hasan Yahya - 22L-7971)
Rows: 100
Cols: 100
Cell size: 6
Apply Grid
Start
Pause
Step
Clear
Randomize
Speed:
Gen: 0
Alive: 2055
Density (0-1): 0.2
Boundary:
Periodic (wrap-around)
Finite (no wrap)
Tip: Click or drag to toggle cells. Use Apply Grid when changing rows/cols/cell size.




We started with about 2055 random alive cells in a grid of 1000 cells (100 x 100) ie around 20 percent alive. In the first 80 steps, the number got reduced till 600 alive cells (most were killed off) but after that it started increasing and the population stayed consistent within the range of 800 to 900 alive cells from 80th step till 200th step (neither increasing too much not decreasing). However, some patterns kept repeating and some were self sufficient (staying exactly the same).

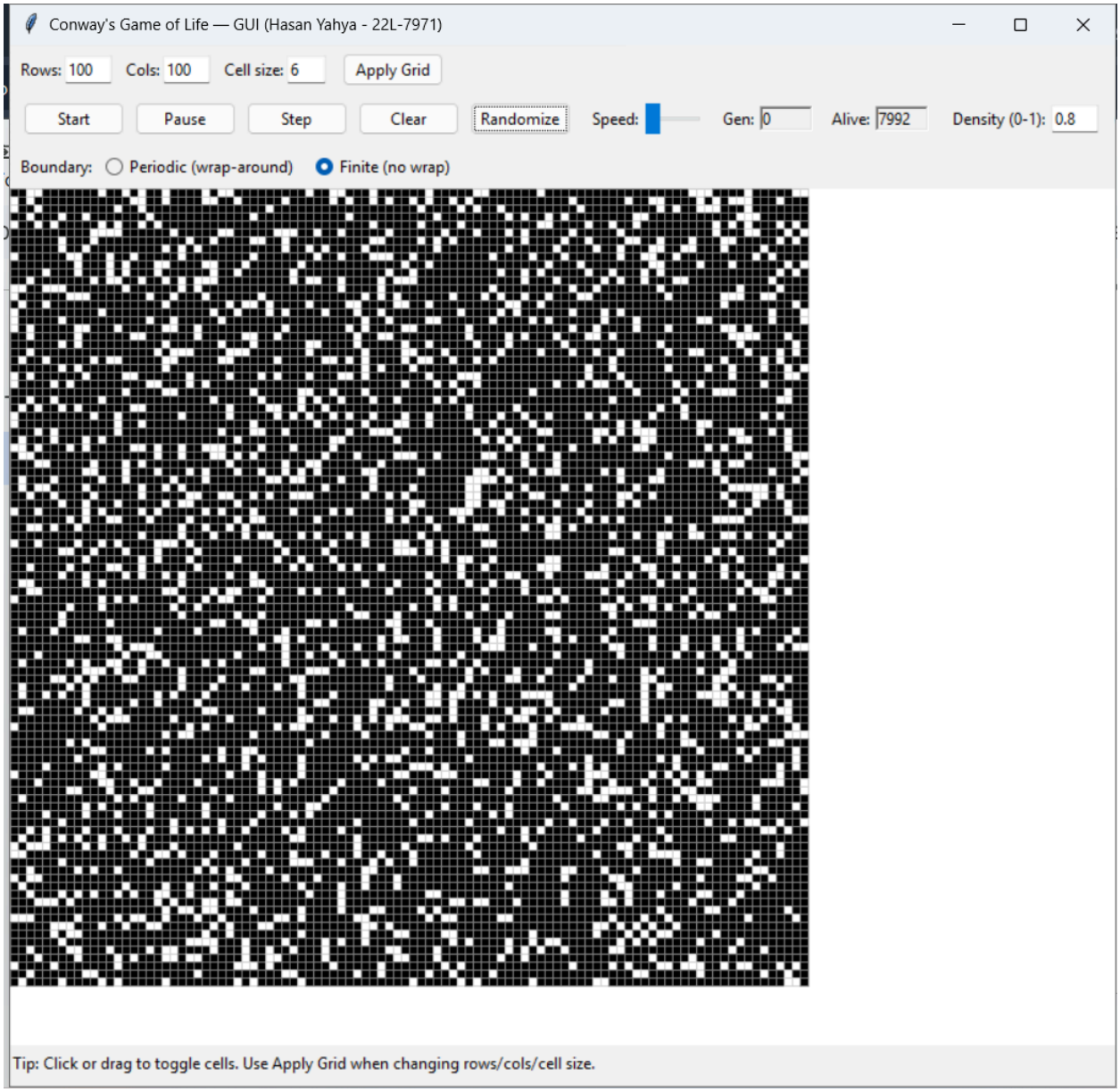
### 6. Finite (No Wrap Around with 100 x 100 Grid) & 0.5 Density:

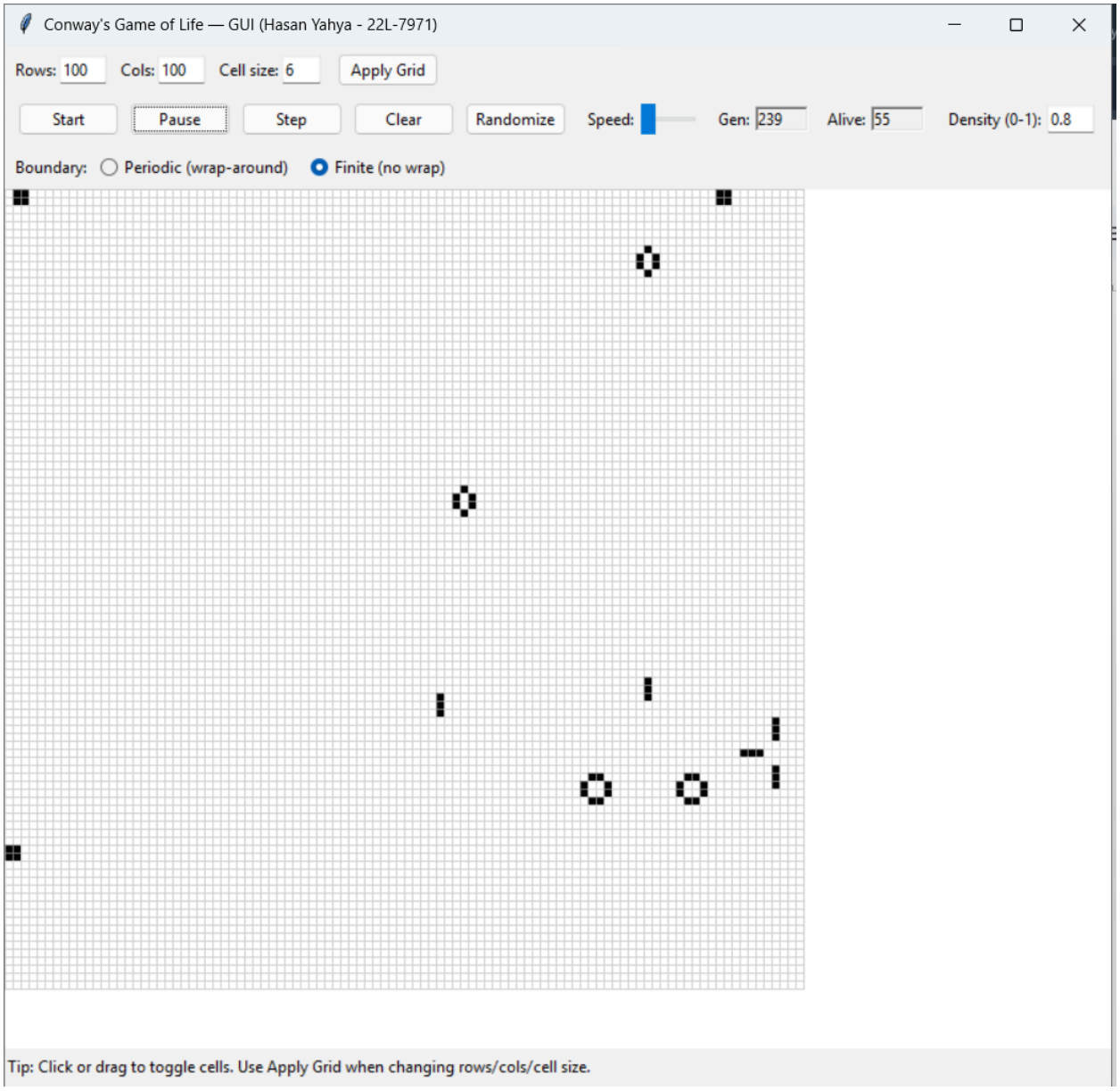

Conway's Game of Life — GUI (Hasan Yahya - 22L-7971)
Rows: 100
Cols: 100
Cell size: 6
Apply Grid
Start
Pause
Step
Clear
Randomize
Speed:
Gen: 0
Alive: 4931
Density (0-1): 0.5
Boundary:
Periodic (wrap-around)
Finite (no wrap)
Tip: Click or drag to toggle cells. Use Apply Grid when changing rows/cols/cell size.




Started with 4931 alive cells, Within the first 10 steps the population of alive cells got halved to 2000. By the 50th step it remained in the 1500 to 2000 range but by the 120th step it was reduced to 600 to 900 alive cell range. So, first it was reduced exponentially then it stayed consistent until being reduced again and at the end remained in the 600 to 900 range (never completely died off).

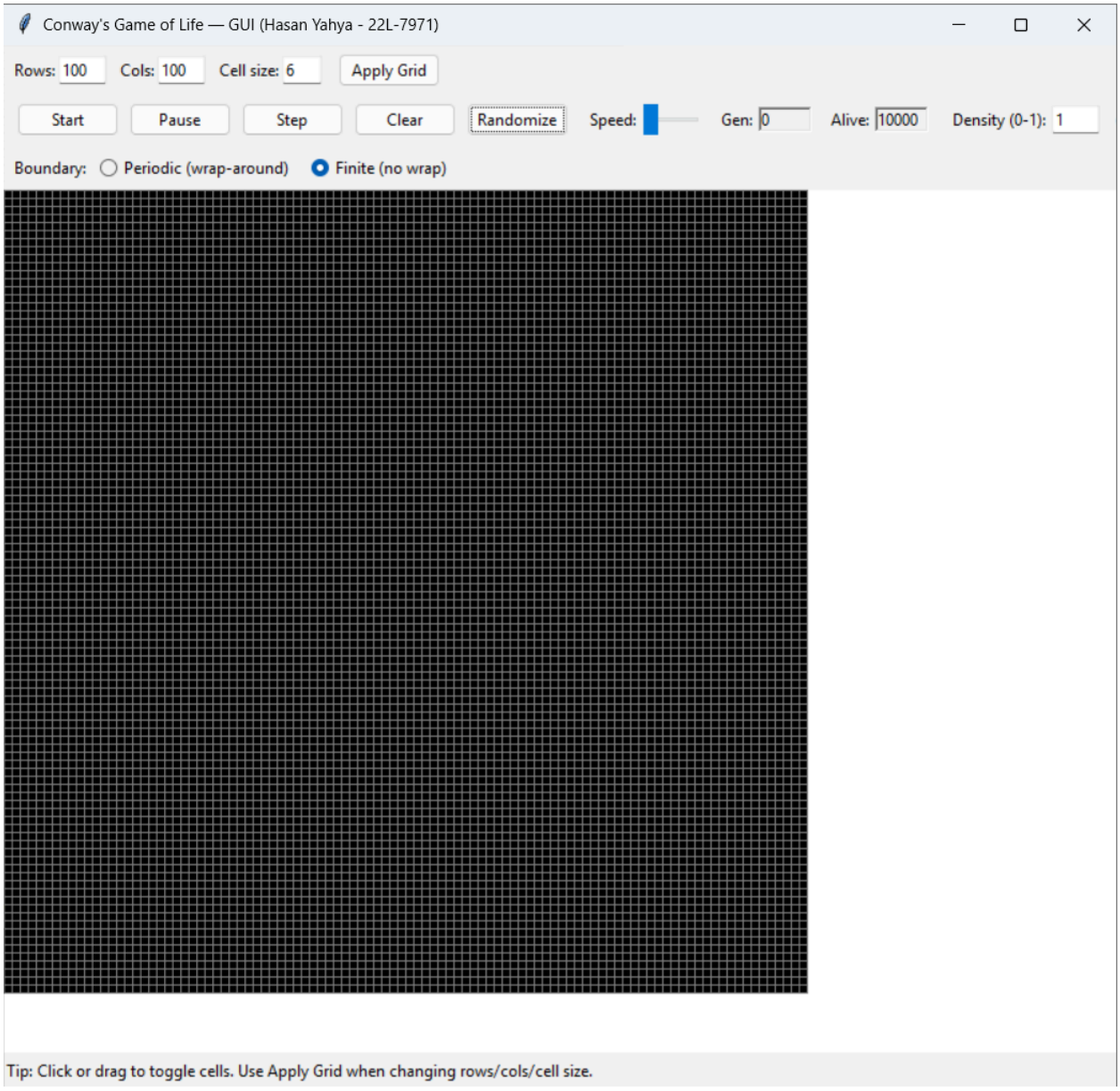
### 7. Finite (No Wrap Around with 100 x 100 Grid) & 0.8 Density:

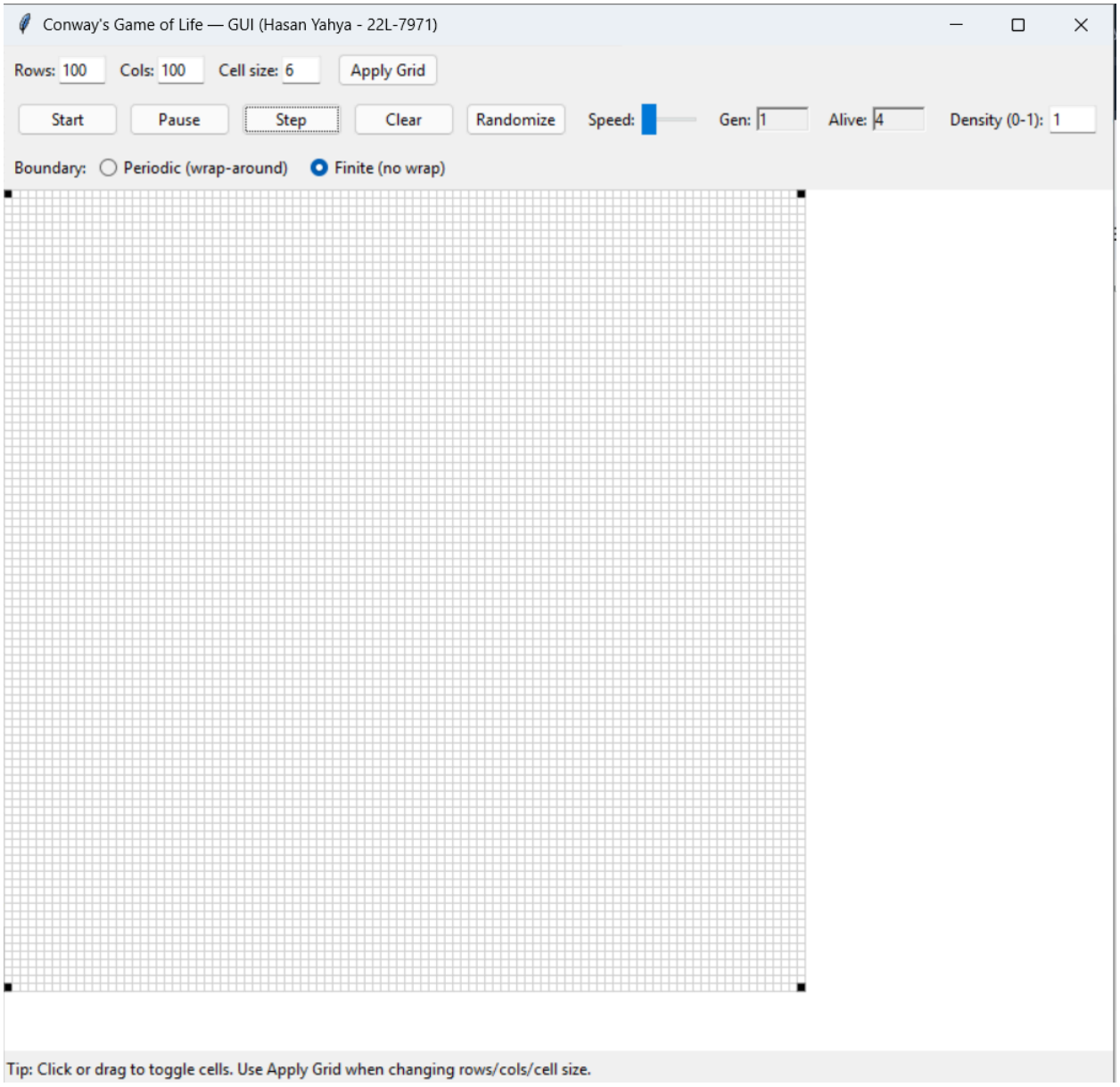

Conway's Game of Life — GUI (Hasan Yahya - 22L-7971)
Rows: 100
Cols: 100
Cell size: 6
Apply Grid
Start
Pause
Step
Clear
Randomize
Speed:
Gen: 0
Alive: 7992
Density (0-1): 0.8
Boundary:
Periodic (wrap-around)
Finite (no wrap)
Tip: Click or drag to toggle cells. Use Apply Grid when changing rows/cols/cell size.

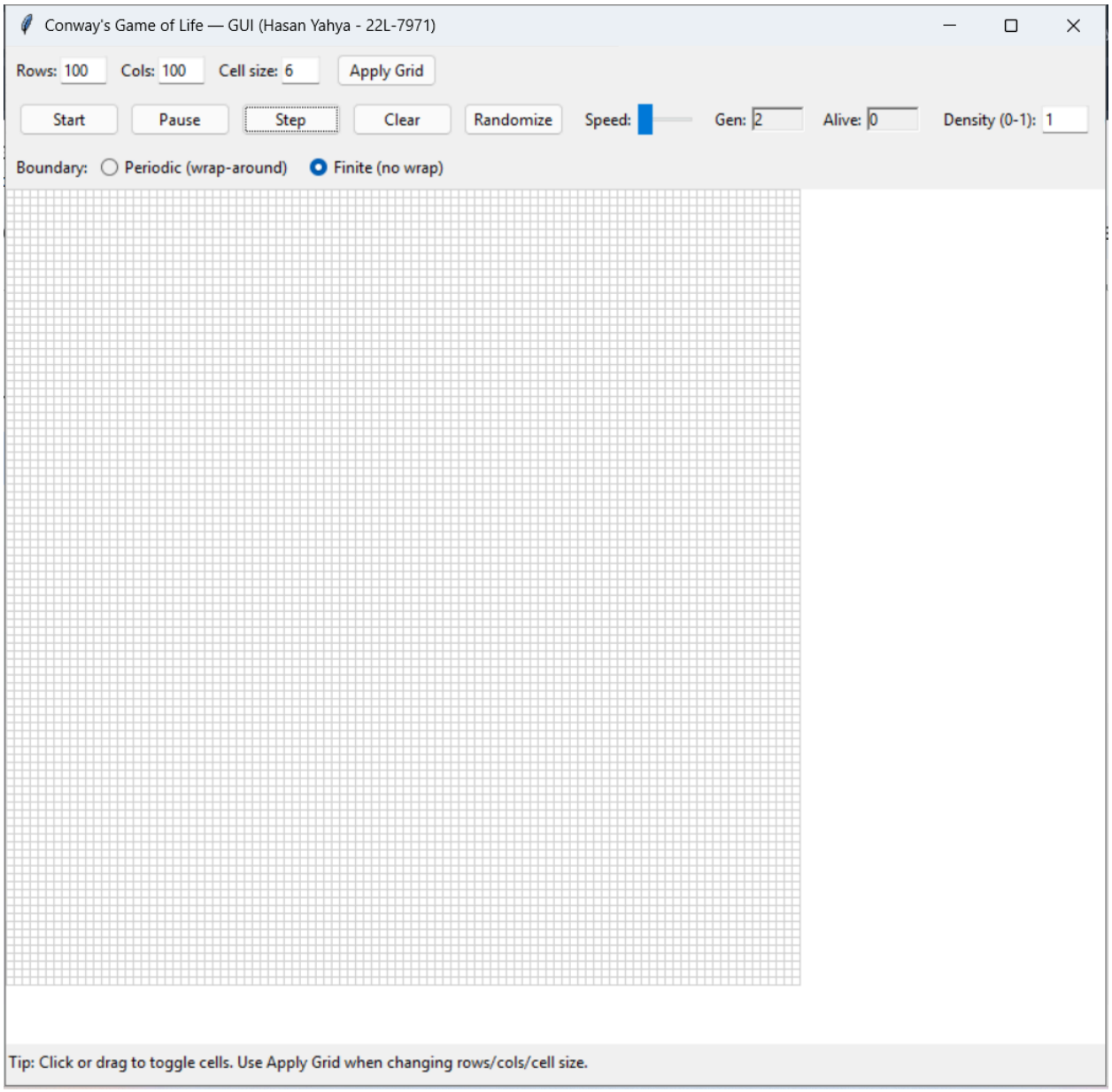



By the 3rd step only 55 alive cells remained and they kept repeating their pattern. Only the self sustainable patterns were allowed to exist. The rest were killed off. This leads to a phenomenon of peace, where each alive group stays in its own little island and doesn't interfere with others and in turn others also don't interfere with it and allows for co-existence.

## 8. Finite (No Wrap Around with 100 x 100 Grid) & 1.0 Density:


Conway's Game of Life — GUI (Hasan Yahya - 22L-7971)
Rows: 100
Cols: 100
Cell size: 6
Apply Grid
Start
Pause
Step
Clear
Randomize
Speed:
Gen: 0
Alive: 10000
Density (0-1): 1
Boundary:
Periodic (wrap-around)
Finite (no wrap)
Tip: Click or drag to toggle cells. Use Apply Grid when changing rows/cols/cell size.


Conway's Game of Life — GUI (Hasan Yahya - 22L-7971)
Rows: 100
Cols: 100
Cell size: 6
Apply Grid
Start
Pause
Step
Clear
Randomize
Speed:
Gen: 1
Alive: 4
Density (0-1): 1
Boundary:
Periodic (wrap-around)
Finite (no wrap)
Tip: Click or drag to toggle cells. Use Apply Grid when changing rows/cols/cell size.




In the 2nd step, only 4 corners remained (unlike in wrap-around, in which in 2nd step all were killed). By third step all were killed in Finite as well.

## Results of the first Run (all 8):

1. **Finite vs Periodic:**
    a. **Finite:** tends to preserve more activity for longer in many random setups because patterns can wrap and interact across the boundaries; oscillators and traveling structures (gliders) may persist and interact indefinitely, causing later detection of repetition or continued activity.

    b. **Periodic:** often leads to more rapid die-out at edges (cells at boundary have fewer neighbors), so extinction or small stable patterns are more common earlier.
2. **Effect of Density:**
    a. **0.2 (20 Percent):** Leads to a gradual/linear decrease of population till it reaches to 7 percent (ie the equilibrium rate) after which it remains almost the same.
    b. **0.5 (50 Percent):** Leads to a Quick/Exponential decrease of population till it reaches to 7 percent (ie the equilibrium rate) after which it remains almost the same.
    c. **0.8 (50 Percent):** Leads to an Instant decrease of population till it reaches to <1 percent. This is similar to simulation of a Disaster and the survivors then live in their own small islands cut off from the rest of the other islands.
    d. **1.0 (100 Percent):** Leads to an instant death of all the people. No one is alive to see another day because of severe overpopulation.

This gives the macro view of Game of Life. Moreover, we don't know the result of increasing or decreasing the Grid Size. For Micro View we should run another test in which we check the densities of 0.2, 0.3 and 0.4 to see how small increases affect the **Cellular Automata** instead of large changes (and is there any difference).

## Density based Simulation - Micro View (Test 2):

By increasing the max steps to 1000 and the grid size to both 100 x 100 and 200 x 200. ***We get the results as given in the Zip File submitted on Google Classroom (so that all the photos don't increase PDF Size to 35 Pages), please check the photos and results from there.***

1. **Grid Size:** Increasing the Grid size simply slows down the inevitable. The result will still be the same. You will just need way more steps to get the result.
2. **Low density (0.2):** many runs die out or settle to stable small clusters quickly. Extinction and stability rates are higher.
3. **Medium density (0.3)**: mixture of both, some runs form oscillators or persistent regions, some die out.
4. **Higher density (0.4)**: more immediate overcrowding; many initial cells die off quickly but also more opportunity for complex interactions; you often see larger transient peaks in population and then fragmentation into oscillators/still life.

The Zip File contains the complete data, terminology used in the Excel Files (in the Zip File) are as under:

1. **Extinction:** population reaches zero.
2. **Stability:** Grid is identical to the previous step (fixed point). Ie all steps are the same.
3. **Oscillation:** a previously seen grid state repeats.
4. **Oscillation Period**: Steps after which it is repeated.

Finalized summary is as below:

| Size | Mode | Density | Oscillation Rate | Mean Population |
|---|---|---|---|---|
| 100 x 100 | Finite | 0.2 | 0.8 | 629 |
| 100 x 100 | Finite | 0.3 | 0.6 | 597 |
| 100 x 100 | Finite | 0.4 | 0.4 | 633 |
| 100 x 100 | Periodic | 0.2 | 0 | 615 |
| 100 x 100 | Periodic | 0.3 | 0.2 | 658 |
| 100 x 100 | Periodic | 0.4 | 0 | 663 |
| 200 x 200 | Finite | 0.2 | 0 | 2415 |
| 200 x 200 | Finite | 0.3 | 0 | 2374 |
| 200 x 200 | Finite | 0.4 | 0 | 2454 |
| 200 x 200 | Periodic | 0.2 | 0 | 2494 |
| 200 x 200 | Periodic | 0.3 | 0 | 2565 |
| 200 x 200 | Periodic | 0.4 | 0 | 2590 |

- As seen above some of the tests had oscillation while others (with 0) did not have any.
- None of the tests went extinct.
- None of the tests were completely stable.

***The above tests (test 1 for macro and 2 for micro that is in the Zip File) suggest that from 0.2 to 0.4 density oscillation decreases and from 0.5 to 0.85 tests go into stable state and if we go higher to 1 (the population goes completely extinct). Make sure to check the Zip File for Indepth tests and photos and data for test 2 (micro).***

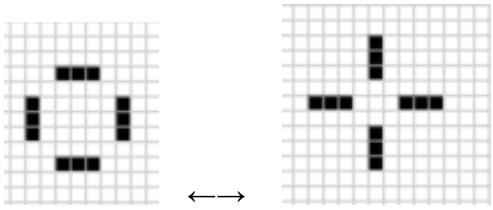
## Patterns Discovered:

Some patterns were discovered during the simulation process which can be divided into 2:

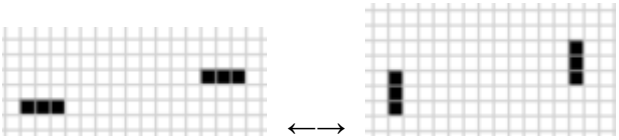
1. **Repeating/Oscillating Patterns:** Those that switch between two states. From one to next and then back in each step.
2. **Self-Sufficient/Stable Patterns:** Those that if no external population (alive) acts, will always stay in the same place and in the same form.
3. **Gliders:** That move (in a group) in one direction (even if there is no external influence) and never stop. They are also self-sufficient but aren't fixed in a place.

Examples for these are given as under:

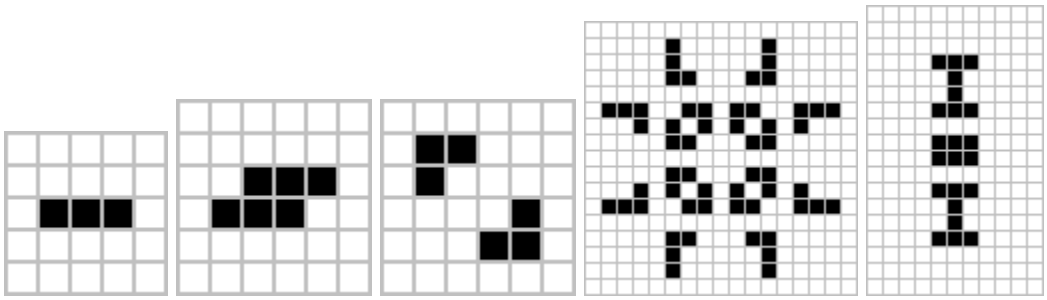
## Repeating/Oscillating Patterns:



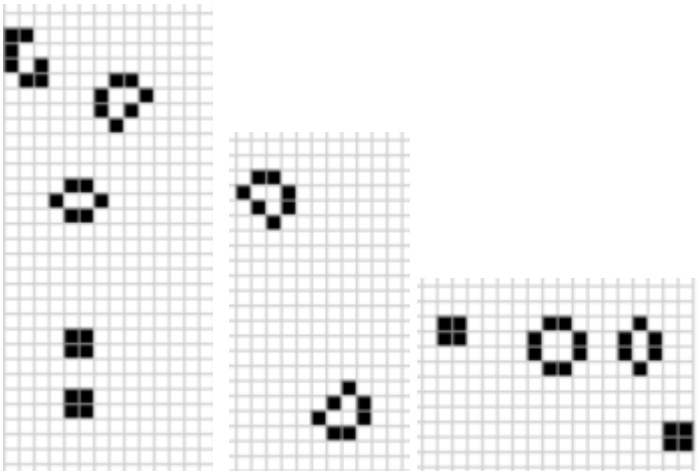
The pattern on the left will always turn into the one on the right, as long as other neighbours are dead (not alive). Other Examples include a line of 3 alive cells (straight line):



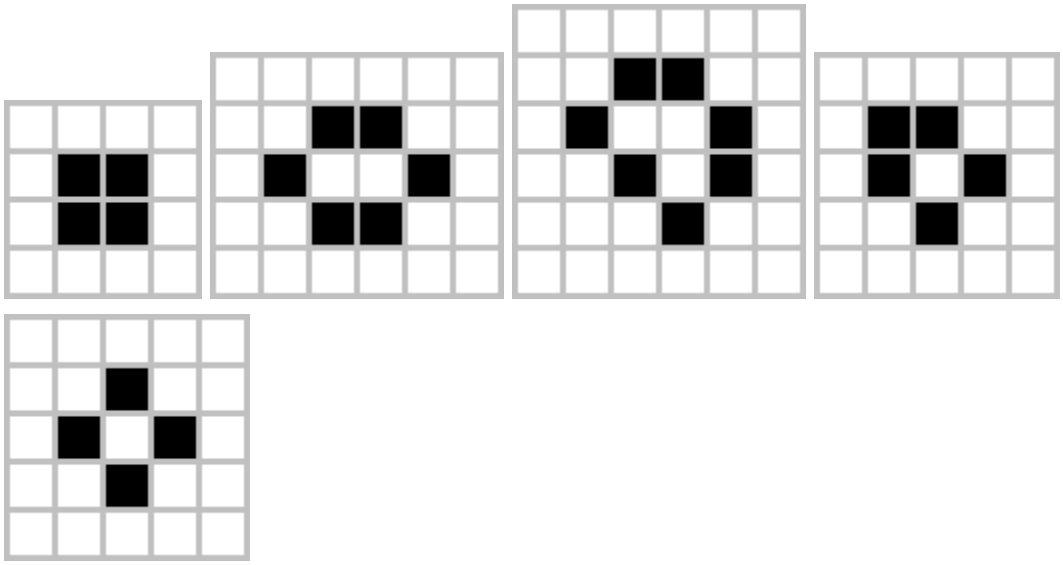
More examples are:



## Self-Sufficient/Stable Patterns:

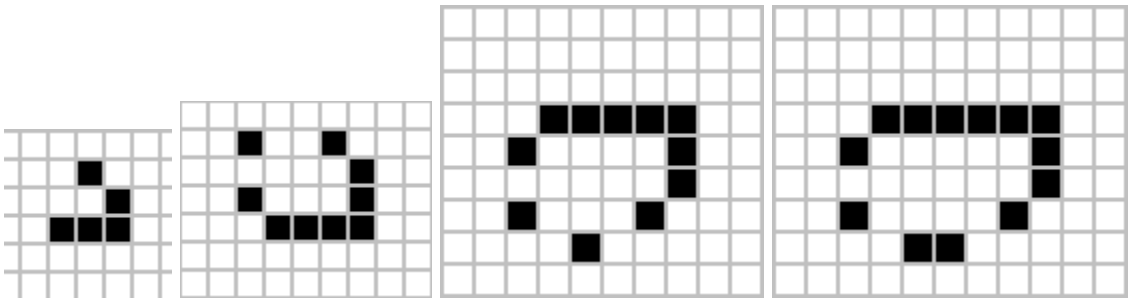


Other examples are:



These will stay exactly the same if other neighbours are dead.

## Gliders:

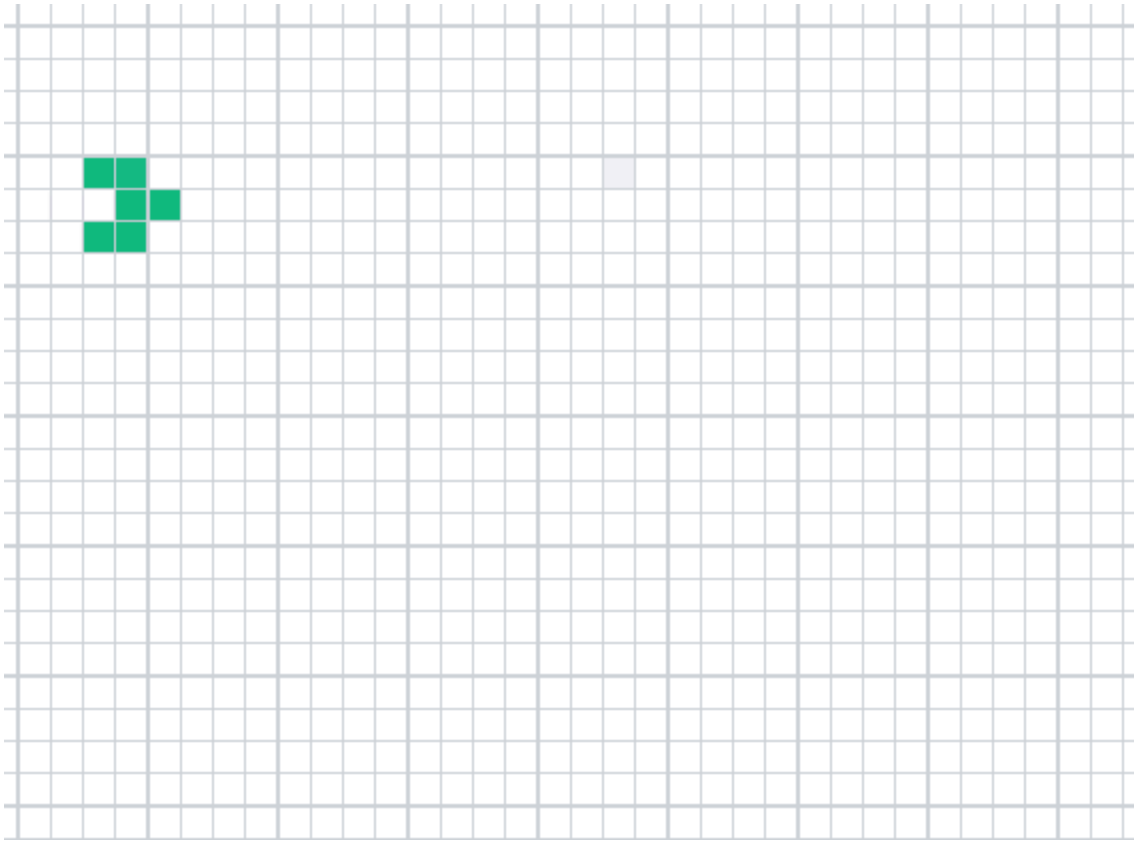


Gliders are also technically repeating, but they don't stay in one place ie keep moving.

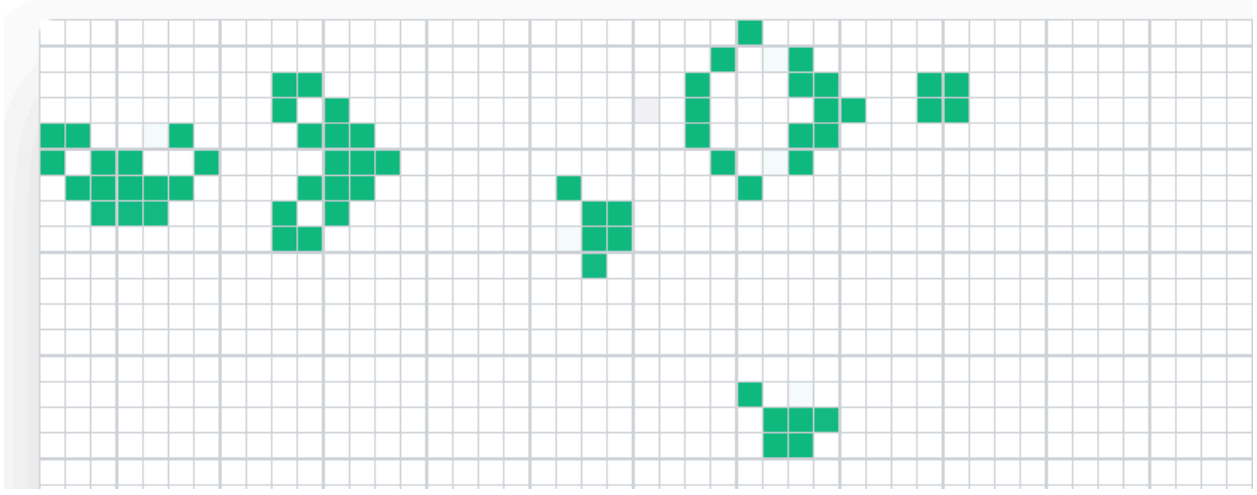
## Other Interesting Patterns:

Some of the other interesting patterns are as below:

1. An island that moves diagonally (to bottom right):



2. Glider Gun that Produces Gliders indefinitely:



## Summary:

In all simulations, the Game of Life behaved very differently depending on the boundary rules. Finite grids slowly lost activity because patterns near the edges died out, leading to small stable shapes or simple loops. Periodic grids stayed much more active since the wrap-around edges let patterns move freely without disappearing, creating more interactions and longer-lasting structures. Lower densities usually died out faster, while higher densities produced busy patterns before eventually settling. Overall, periodic boundaries helped the system stay alive and complex, while finite boundaries caused the grid to become quiet much sooner.

---