

2022-01-20

# Gra w życie



Tomasz Wierzbicki

## Spis treści

1. Cel tematu .....	2
2. Opis problemu .....	2
3. Metoda wykonania.....	2
4. Wizualizacje .....	3
5. Wnioski .....	6

## 1. Cel tematu

Założeniem tematu było wykonanie symulacji automatu komórkowego „Gra w życie” wraz z wizualizacją graficzną. Początkowa kolonia bakterii ma być losowana. Ilość iteracji ma być parametrem programu.

## 2. Opis problemu

„Gra w życie” jest jednym z pierwszych przykładów automatu komórkowego, została wymyślona w 1970 roku przez brytyjskiego matematyka Johna Conwaya. Nie jest grą w powszechnym rozumieniu tego słowa. Sam autor nazywał ją „grą dla zera graczy”. Możemy o niej myśleć jako o symulacji – alegorii życia.

Mamy płaszczyznę, a na niej rozmieszczone obok siebie komórki. Jak na kartce papieru w kratkę. Kratki to poszczególne komórki. Komórki te przyjmują wartości binarne - mogą być martwe (pusta kratka -0) lub żywe (zamalowana komórka-1). Ich żywot jest określony na podstawie życia sąsiadów - każda komórka może mieć maksymalnie 8 sąsiadów (z czterema graniczy ścianami, a z czterema łączy się rogami). Życie płynie, w przypadku naszej gry – w rundach zwanych krokami.

W każdej rundzie wszystkie komórki w grze zmieniają się (lub się nie zmieniają) zgodnie z czterema prostymi zasadami:

1. Żywa komórka, która ma mniej niż dwóch sąsiadów, w kolejnej rundzie umiera.
2. Żywa komórka, która ma więcej niż trzech sąsiadów, umiera.
3. Żywa komórka, która ma dwóch lub trzech sąsiadów – przeżywa.
4. Martwa komórka, która ma trzech żywych sąsiadów – ożywa.

Te cztery zasady stanowią bardzo uproszczoną alegorię zasad rządzących życiem w jego różnych przejawach od organizmów jednokomórkowych po społeczeństwa.

## 3. Metoda wykonania

W celu realizacji projektu zastosowałem narzędzie jakim jest MATLAB. Aby rozwiązać problem, podczas implementacji posłużyłem się 2 macierzami, aby nie nadpisywać powstałych wyników. Na samym początku w zmiennej „steps” deklarujemy ile rund ma trwać symulacja gry oraz rozmiar macierzy. Pierwsza macierz zawiera wygenerowane losowo wartości w przedziale od 0 do 1, po czym zaokrąglam je do pełnych wartości. Drugą macierz wypełniam zerami. Zadeklarowane rozmiary obu macierzy zwiększam o 2, aby stworzyć swego rodzaju „ramkę”. Ramkę tworzę w celu uproszczenia kodu. Dzięki niej, algorytm zliczający sumę sąsiadów danej komórki ma tę samą postać niezależnie od tego, czy komórka znajduje się w rogu, na krawędzi, czy wewnątrz macierzy. Bez użycia ramki, przed obliczeniem sumy kod wymagałby sprawdzenia, gdzie znajduje się dana komórka i suma musiałaby być liczona z pominięciem niektórych komórek sąsiadujących, aby nie wyjść poza macierz. Określając odpowiednio położenie j oraz k dostaję się do każdej sąsiadującej komórki:

```

suma = board1(j-1, k-1) + board1(j-1, k) + board1(j-1, k+1) ...
      + board1(j, k-1) + board1(j, k+1) ...
      + board1(j+1, k-1) + board1(j+1, k) + board1(j+1, k+1);

```

Po zliczeniu wszystkich żywych sąsiadów realizuję fundamentalne zasady „Gry w życie”:

```

if board1(j, k) == 1 && (suma == 2 || suma == 3)
    board2(j, k) = 1;
elseif board1(j, k) == 0 && suma == 3
    board2(j, k) = 1;
else
    board2(j, k) = 0;

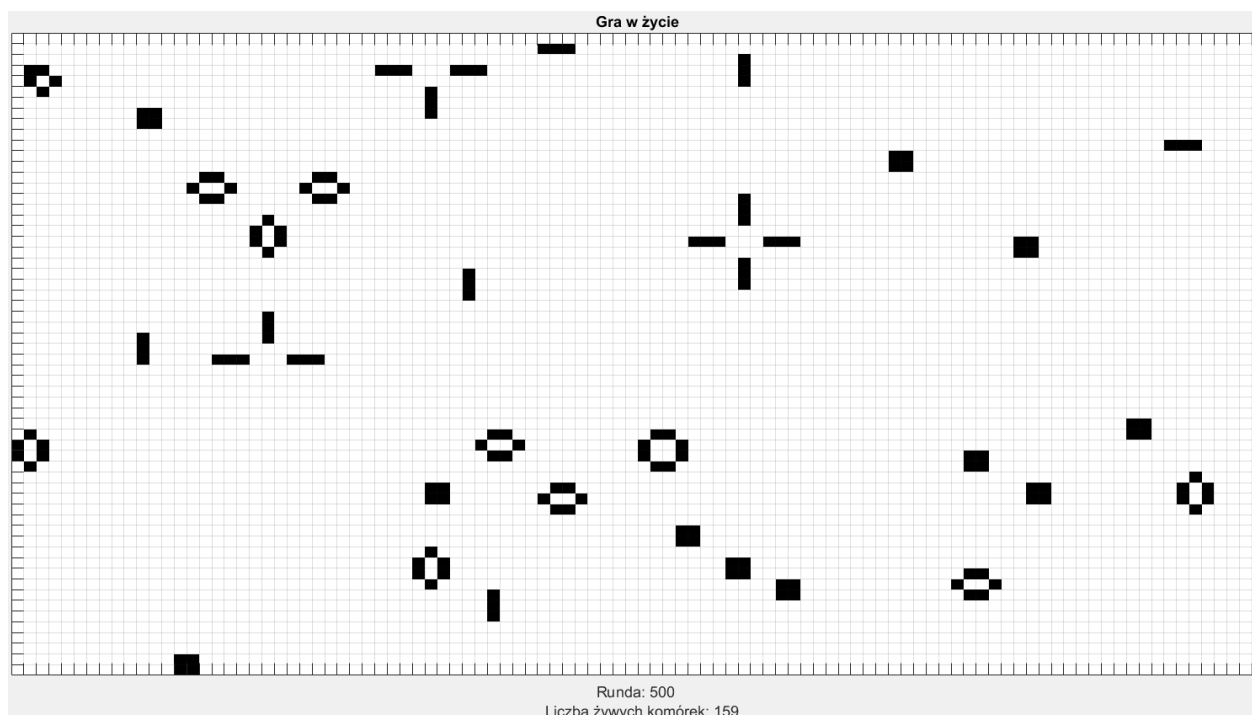
```

W końcowej fazie algorytmu, do macierzy 1 przypisuję wartości z macierzy 2, w której jest zapisany obecny stan gry z bieżącej rundy. Po wykonaniu tego kroku ponownie zeruję 2 macierz w celu uniknięcia niepotrzebnych komplikacji w kolejnych rundach.

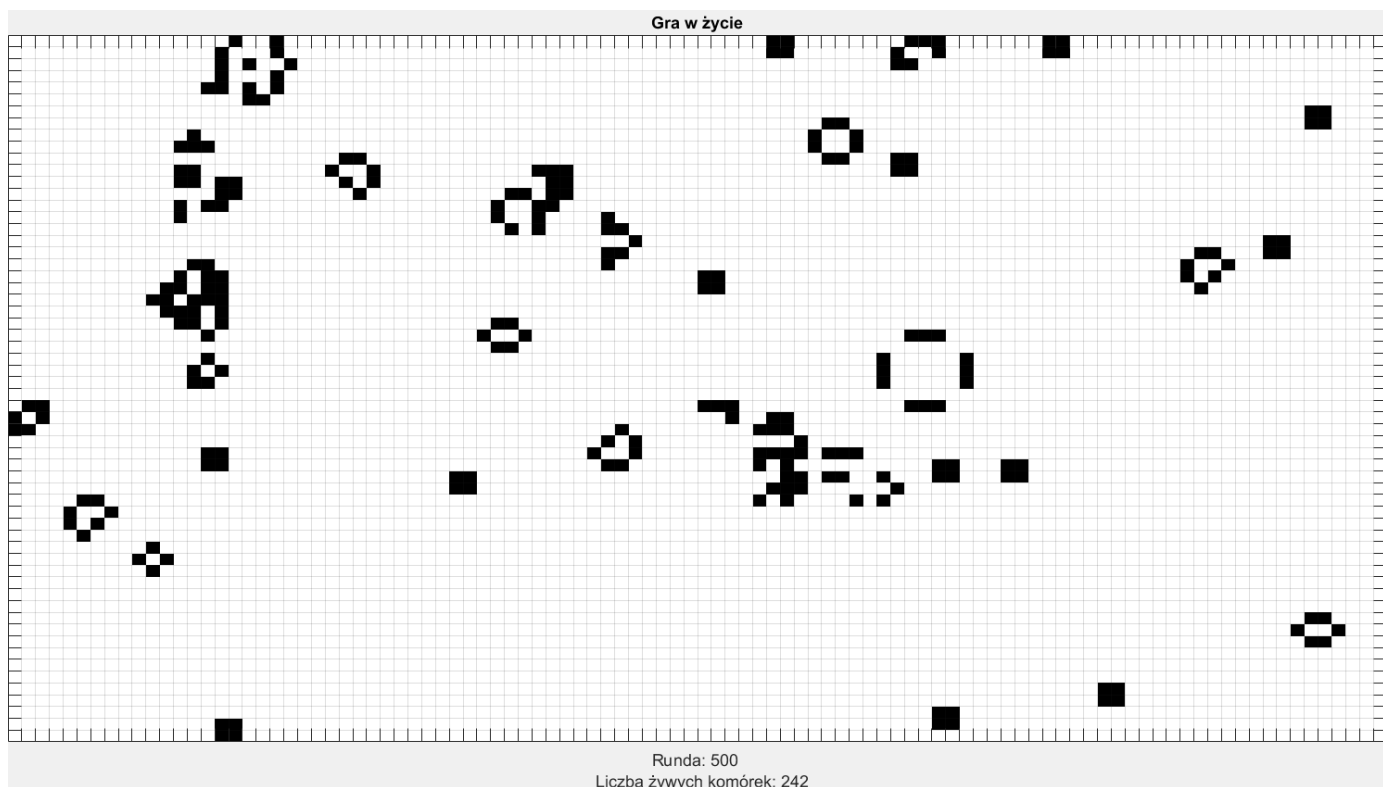
#### 4. Wizualizacje

Poniżej znajdują się wizualizacje „Gier w życie” przy różnych ilościach zadeklarowanych rund.

Poniższe 2 „gry” zostały wykonane dla 500 iteracji (rund).

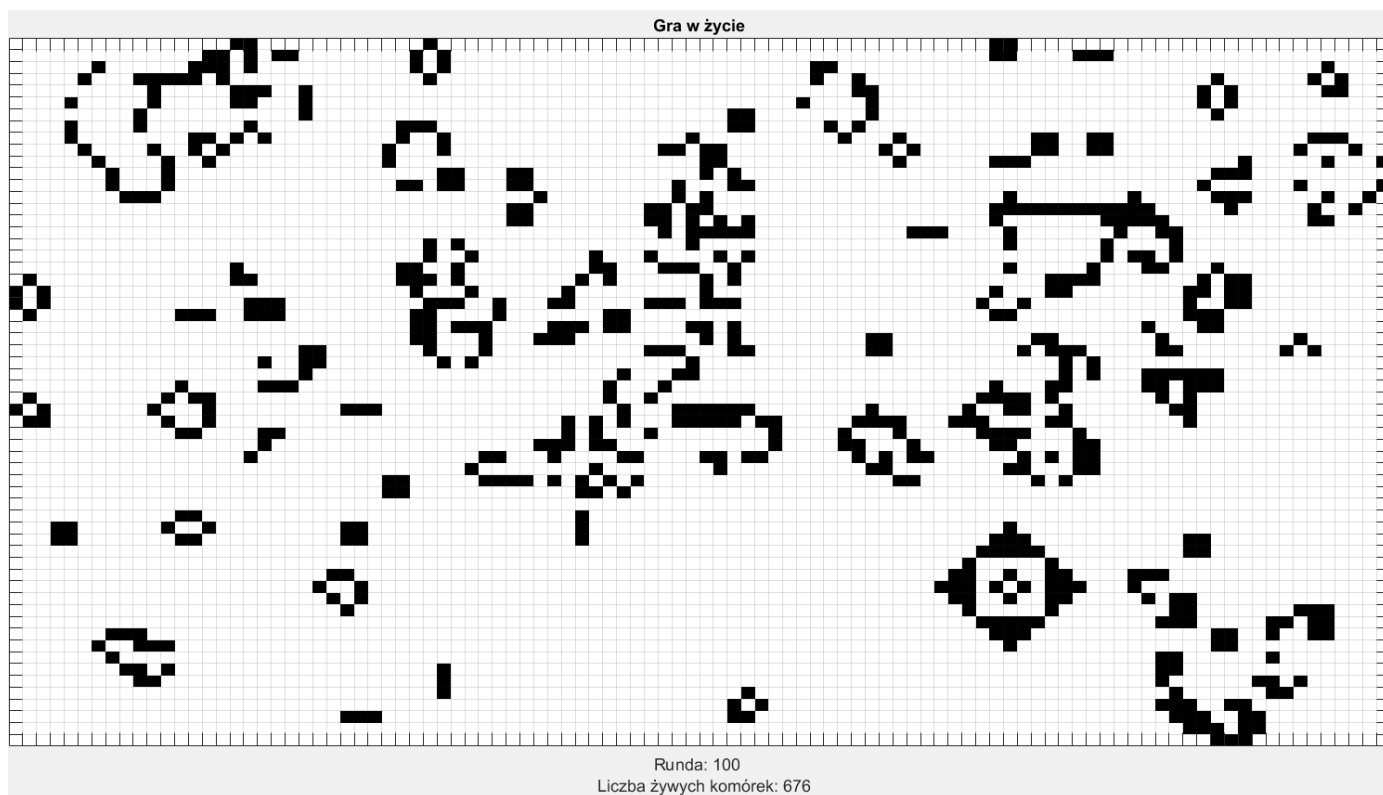


Rysunek 1: Pierwsza gra trwająca 500 rund

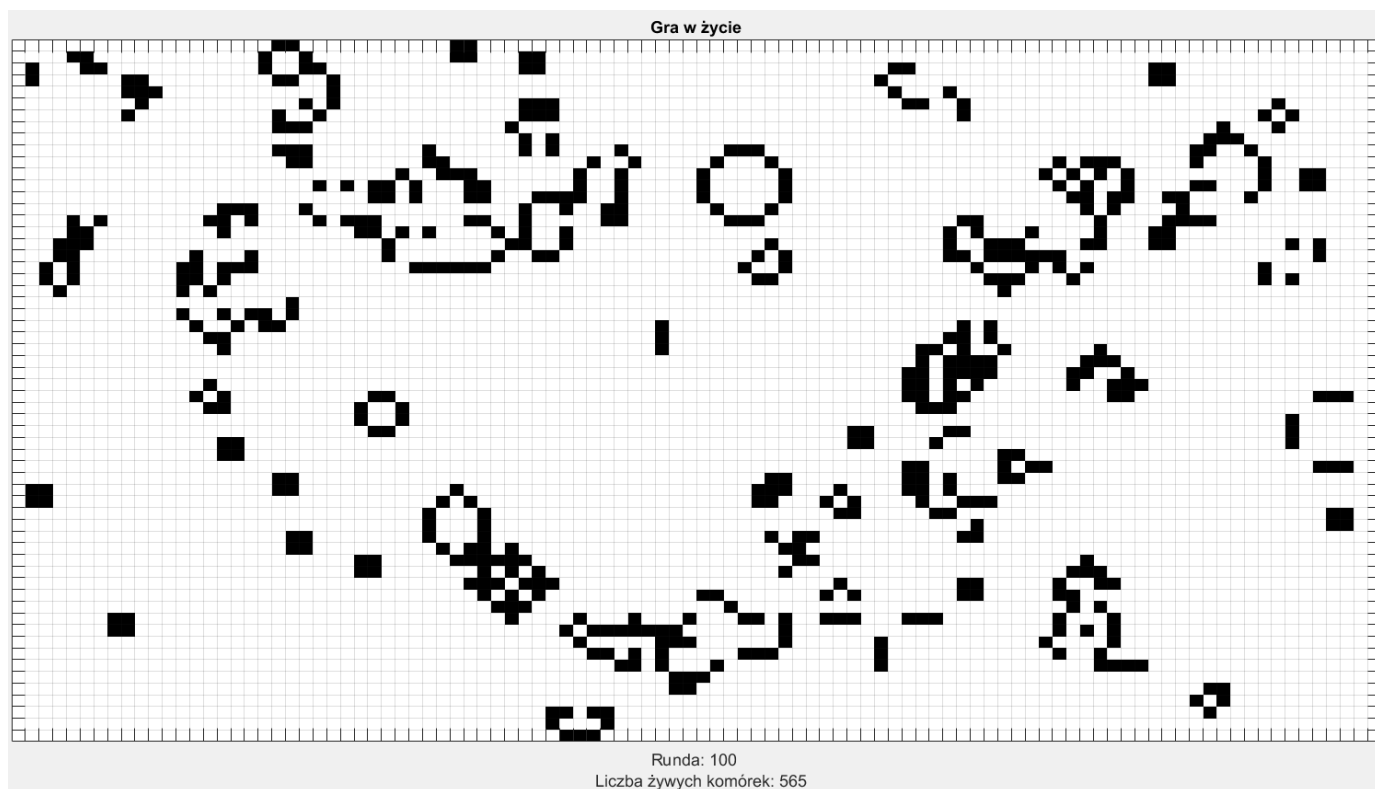


*Rysunek 2: Druga gra trwająca 500 rund*

Poniższe 2 „gry” zostały wykonane dla 100 iteracji (rund).

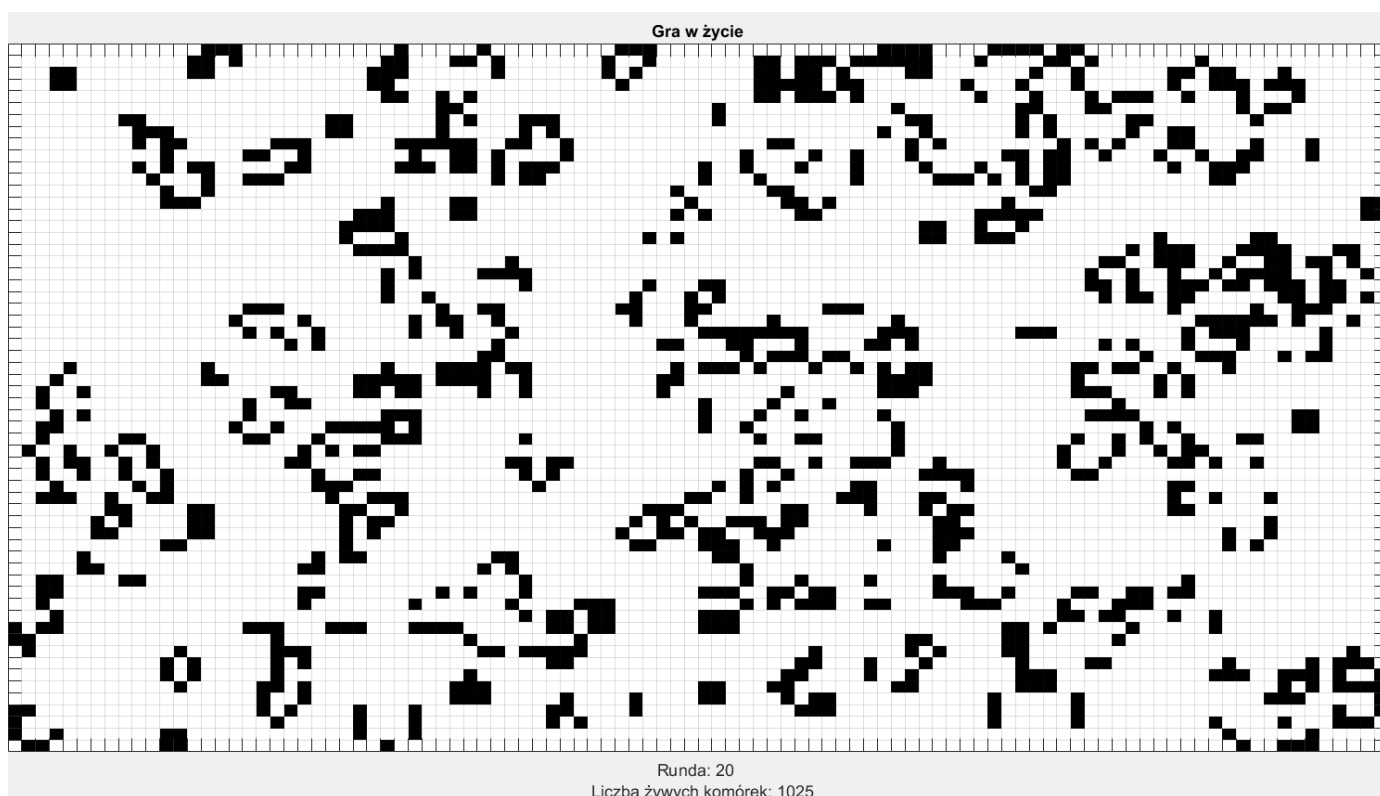


*Rysunek 3: Pierwsza gra trwająca 100 rund*

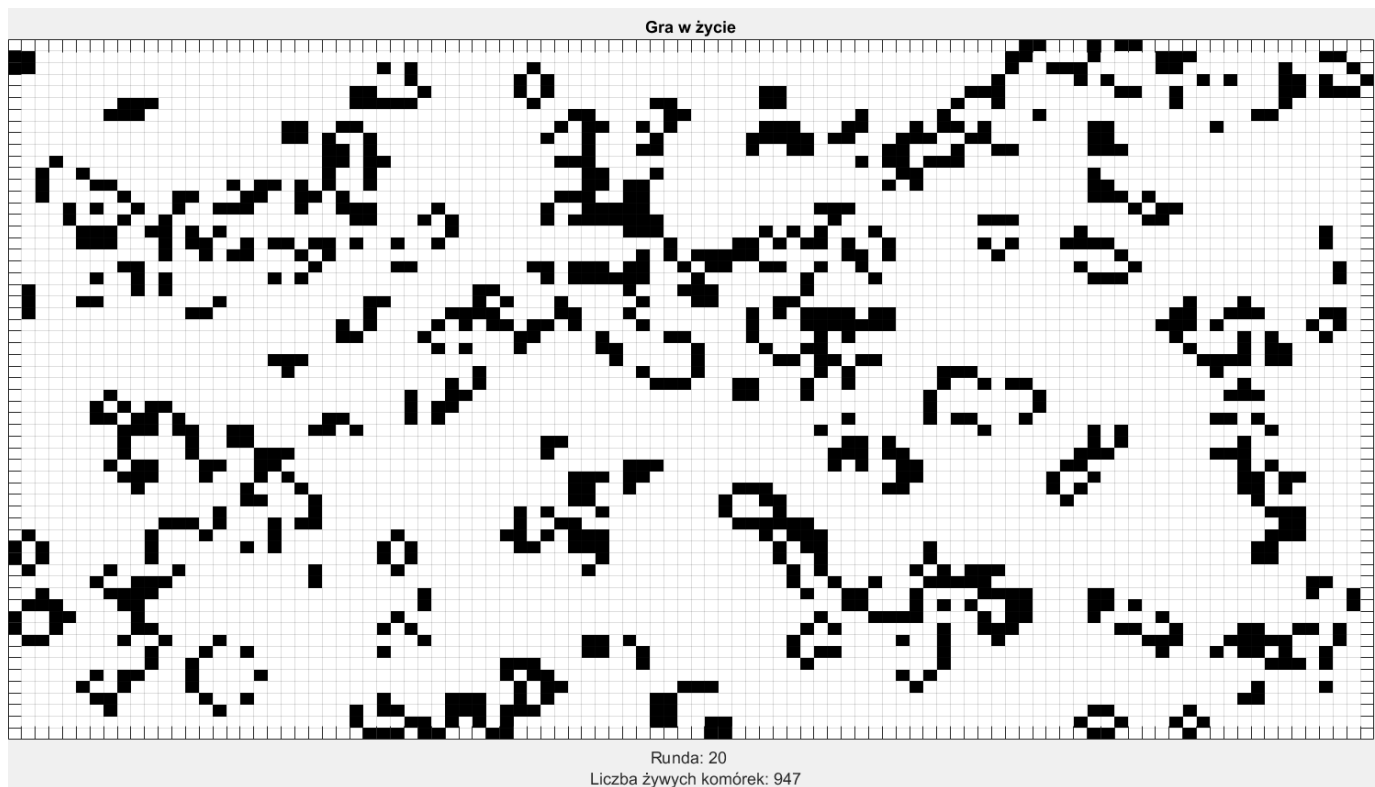


*Rysunek 4: Druga gra trwająca 100 rund*

Poniższe 2 „gry” zostały wykonane dla 20 iteracji (rund).



*Rysunek 5: Pierwsza gra trwająca 20 rund*



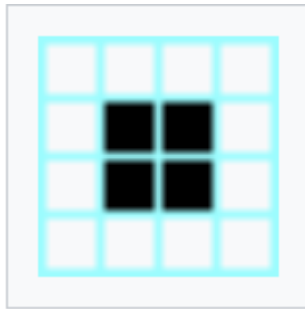
*Rysunek 6: Druga gra trwająca 20 rund*

## 5. Wnioski

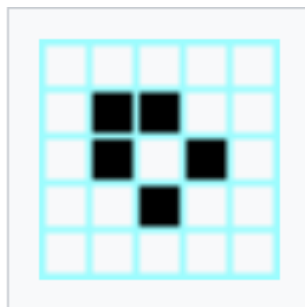
„Gra w życie” jest automatem komórkowym, który symuluje przebieg życia cywilizacji przy stosowaniu się do 4 kluczowych zasad. Po zaobserwowaniu przebiegu życia bakterii przy poszczególnych ilościach iteracji, można dostrzec, że stan gry stabilizuje się wraz z większą ilością rund. Po 20 rundach na planszy panuje większy chaos oraz widnieje dużo więcej żywych komórek, które są jeszcze nieustabilizowane, niż po zadeklarowaniu 100 czy 500 rund. Po przebiegu 500 rund gry możemy zaobserwować dużo mniejszą liczbę komórek, które przeżyły i są one ustrukturyzowane. Przy 100 rundach można zauważyć, że pojawiają się schematy niezmiennych struktur, ale występują również nieuporządkowane elementy żywych bakterii -struktury niestałe.

Struktury niezmiennie, inaczej stabilne lub statyczne, pozostają identyczne bez względu na krok czasowy (dla każdej żywej komórki spełniony jest warunek przetrwania i dla żadnej spośród martwych nie jest spełniony). Najprostsza taka struktura (blok) składa się z 4 żywych komórek. Pojawiają się bardzo często jako produkty końcowe ewolucji struktur.

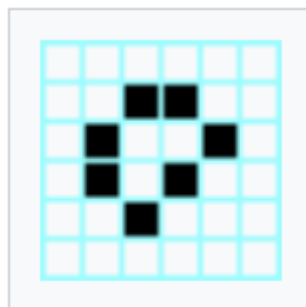
Przykłady występowania statycznych struktur:



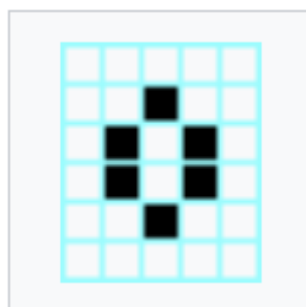
*Rysunek 7: Blok*



*Rysunek 8: Łódź*

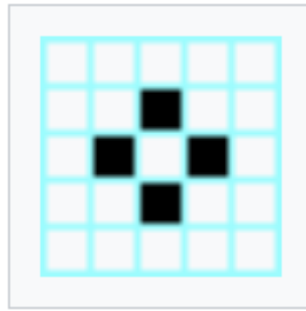


*Rysunek 9: Bochenek*

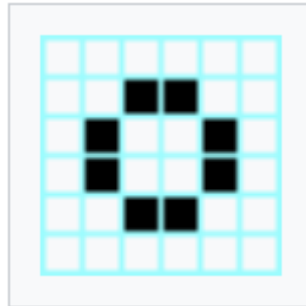


*Rysunek 10: Kryształ*





*Rysunek 11: Koniczynka*



*Rysunek 12: Staw*

W symulacji występują również struktury, które zmieniają się okresowo i co pewien czas powracają do swojego stanu pierwotnego – oscylatory. Najprostsza taka struktura składa się z trzech żywych komórek położonych w jednym rzędzie (blinker). Najprostsze z nich dość często pojawiają się jako produkty końcowe ewolucji struktur.