

## Gra w życie

jeden z pierwszych i najbardziej znanych przykładów automatu komórkowego, wymyślony w roku 1970 przez brytyjskiego matematyka Johna Conwaya. Gra została spopularyzowana przez Martina Gardnera na łamach Scientific American. Od momentu publikacji zawsze wzbudzała duże zainteresowanie z powodu zaskakującego sposobu, w jaki struktury potrafią ewoluować. To właśnie jej pojawienie się wzbudziło zainteresowanie automatami komórkowymi wśród studentów, którzy traktowali ją jako rozrywkę oraz fizyków, którzy zwrócili uwagę na możliwości automatów w zakresie symulatorów fizycznych. Dzisiaj matematyków, ekonomistów i naukowców z innych dziedzin interesuje sposób, w jaki przy zastosowaniu tylko kilku prostych reguł powstają skomplikowane struktury.

Gra toczy się na nieskończonej planszy (płaszczyźnie) podzielonej na kwadratowe komórki. Każda komórka ma ośmiu „sąsiadów” (tzw. sąsiedztwo Moore’a), czyli komórki przylegające do niej bokami i rogami. Każda komórka może znajdować się w jednym z dwóch stanów: może być albo „żywa” (włączona), albo „martwa” (wyłączona). Stany komórek zmieniają się w pewnych jednostkach czasu. Stan wszystkich komórek w pewnej jednostce czasu jest używany do obliczenia stanu wszystkich komórek w następnej jednostce. Po obliczeniu wszystkie komórki zmieniają swój stan dokładnie w tym samym momencie. Stan komórki zależy tylko od liczby jej żywych sąsiadów. W *grze w życie* nie ma graczy w dosłownym tego słowa znaczeniu. Udział człowieka sprowadza się jedynie do ustalenia stanu początkowego komórek.

Program został uruchomiony na 16 procesach z 50 iteracjami.

- Martwa komórka, która ma dokładnie 3 żywych sąsiadów, staje się żywa w następnej jednostce czasu (rodzi się);
- Żywa komórka z 2 albo 3 żywymi sąsiadami pozostaje nadal żywa; przy innej liczbie sąsiadów umiera

Stan początkowy, wynik 1 i 2 iteracji:

```
student@studentVB:~/lab7/Zad1$ mpicc Zad2.c -o Zad2.exe
student@studentVB:~/lab7/Zad1$ mpirun -np 16 --oversubscribe Zad2.exe 50
```

```
Iteration 0: final grid:
```

[illegible]

```
Iteration 1: final grid:
```

[illegible]

Stan w środku, wynik 30 i 31 iteracji:

```
Iteration 29: final grid:
```

[illegible]

```
Iteration 30: final grid:
```

[illegible]

Stan na końcu, wynik 49 i 50 iteracji:

```
Iteration 48: final grid:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0

Iteration 49: final grid:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0
student@studentVB:~/lab7/Zad1$
```

[https://pl.wikipedia.org/wiki/Gra\\_w\\_%C5%BCycie](https://pl.wikipedia.org/wiki/Gra_w_%C5%BCycie)