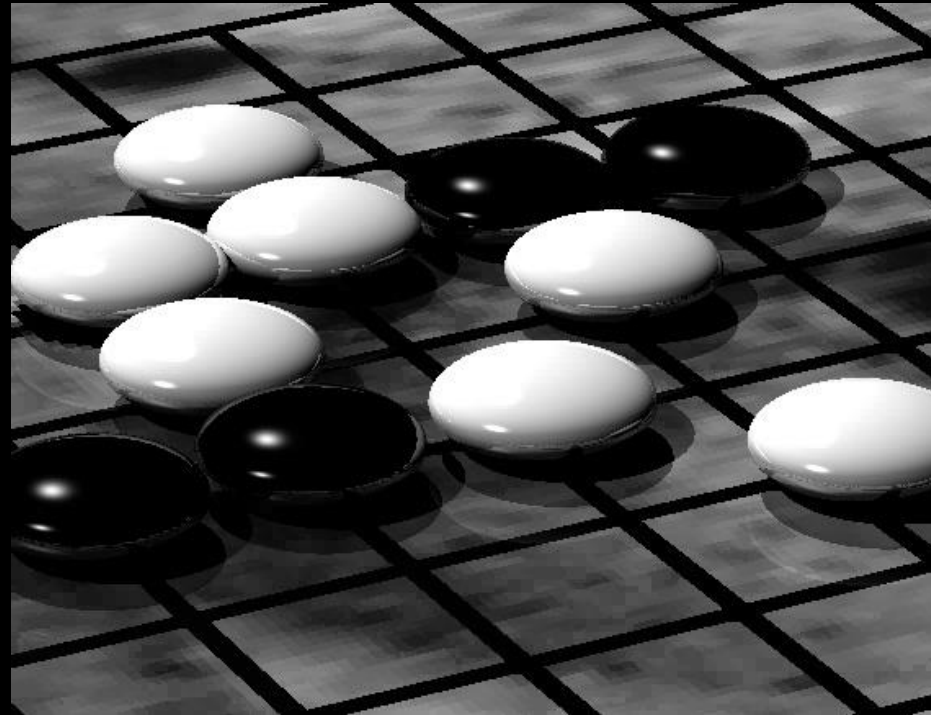


# Symulacje komputerowe w fizyce



17-18.01.2023

Ćwiczenia 13: Automaty komórkowe

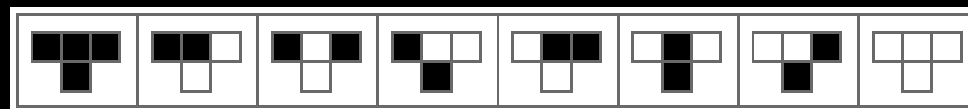
# Zadanie

1. Stwórz kod pozwalający na symulację ewolucji dowolnego automatu komórkowego. Sprawdź działanie programu dla różnych reguł i warunków początkowych.
- 2.\* Rozszerz program o możliwość symulacji automatów odwracalnych. Zbadaj ewolucję automatu 122R dla  $T = 15000$  kroków, używając warunków początkowych z pliku ca122.txt. Zastanów się, jak możliwa jest taka ewolucja.

# Zadanie 1 - szkielet kodu

1. Wczytanie numeru reguły i przetłumaczenie jej na funkcję przejścia (tłumaczenie ma zachodzić automatycznie – program przyjmuje jedynie numer i sam go przelicza na funkcję przejścia).

reguła 150



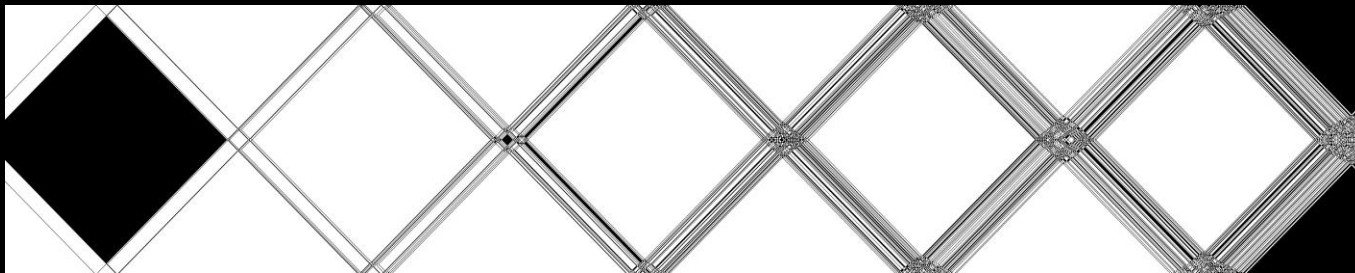
2. Inicjalizacja pierwszego rzędu na dwa możliwe sposoby: losowo (np. rząd o długości  $N = 500$ ) lub  $k$  (np.  $k = 40$ ) środkowych czarnych komórek otoczonych  $L$  ( $L = 460$ ) białych komórek.
3. Algorytm do ewolucji (bez pętli po komórkach!) sprawdzający sąsiedztwo (z periodycznymi warunkami brzegowymi) i stosujący odpowiedni fragment reguły.
4. Pętla po czasie z ewolucją przez  $T = 1000$  kroków czasowych i dodawaniem efektów kolejnych iteracji do tablicy z wynikami.

# Kilka wskazówek

sprawdzanie sąsiadów – przez funkcję np.roll (z automatu PBC)

szybkie rysowanie:

```
from PIL import Image, ImageDraw
img=Image.new("RGB",(width,height),(255,255,255))
draw = ImageDraw.Draw(img)
for y in range(height):
    for x in range(width):
        if data[y][x]: draw.point((x,y),(0,0,0))
img.save("ca.png")
```



# Zadanie 2 – automaty odwracalne

Prosty algorytm Fredkina do konstruowania automatów odwracalnych:

$$c_i(t+1) = \varphi[c_{i-1}(t), c_i(t), c_{i+1}(t)] + c_i(t-1) \mod 2$$

Jak można łatwo udowodnić, taką regułę można przekształcić:

$$c_i(t-1) = \varphi[c_{i-1}(t), c_i(t), c_{i+1}(t)] + c_i(t+1) \mod 2$$

Trzeba zauważyć, że automaty odwracalne wymagają dwóch rzędów warunków początkowych. W zadaniu dodatkowym wczytamy plik z warunkami funkcją `np.loadtxt(file_name, dtype = int)`, a następnie weźmiemy pierwszy z wczytanych rzędów jako początkowe  $c_i(t-1)$ , a drugi jako  $c_i(t)$ .

# Punktowanie zadania:

1. Program do ewolucji zwykłego automatu, działający dla dowolnej reguły – 1 pkt.
- 2.\* Ewolucja automatu 122R z danych warunków początkowych – 0.2 pkt.

# Praktyczne zastosowanie automatów komórkowych...

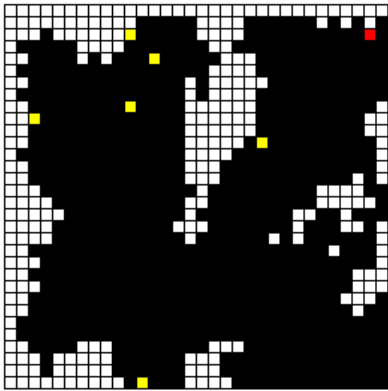
## Procedurally-Generated Dog Simulator

### Instructions

Click or tap anywhere on the map to make the ■ HUMAN visit it.

The ■ DOG likes the human but it likes ■ TREATS even more. It also fears the ■ SKATEBOARDER.

Click or tap just past the edges of the map to explore new areas.



If you'd like a goal, consider finding the dog and leading it to treats.

<http://toolness.github.io/ca-fun/>

