

## Prosta animacja

Aby wykonać animację w bibliotece `turtle` będziemy generować klatki i po kolei je wyświetlać. Przydadzą się nam funkcje:

1. `screen.tracer(0)`
2. `time.sleep()`
3. `screen.update()`
4. `clear()`

Rozważmy następujący przykład.

```
1 import turtle
2 import time
3
4 screen = turtle.Screen()
5 screen.setup(500,500)
6 screen.tracer(0)          # obraz nie pojawia sie automatycznie
7 pen = turtle.Turtle()
8 pen.speed(0)
9 pen.hideturtle()
10
11 def drawSquare() :
12     for _ in range(4) :
13         pen.forward(100)
14         pen.left(90)
15
16 pen.penup()
17 pen.goto(-350, 0)        # na lewy koniec ekranu
18 pen.pendown()
19
20 while True :
21     pen.clear()           # czyszcimy ekran z poprzedniej klatki
22     drawSquare()          # rysuj kwadrat
23     screen.update()       # pokaz nowa klatke
24     pen.forward(0.02)
25     # time.sleep(0.01)    # zwolnij animacje (import time)
26
27 turtle.done()
```

Kod generuje animację kwadratu przesuwanego się od lewej do prawej.

W linii 6 nakazujemy żółwiowi aby rysował do bufora, ale go nie pokazywał dopóki mu nie powiemy, żeby to zrobił (linia 23).

Pamiętajcie, że przed narysowaniem nowej klatki animacji starą trzeba zmasać z ekranu (linia 21).

Jeśli animacja przebiega zbyt szybko możemy np. posłużyć się funkcją `sleep` (z biblioteki `time`), która wstrzyma działanie programu o liczbę sekund podaną jako parametr (linia 25).

## Zadanie

Waszym zadaniem jest zaimplementowanie żółwiem prostej animacji przedstawiającej choinkę. Czym jest choinka?

- trzy zielone (o różnych odcieniach) trójkąty, z których każdy położony wyżej jest mniejszy i nachodzi na swojego poprzednika (patrz rysunki)
- kolorowe bombki/lampki umieszczone losowo na drzewku zmieniające kolory pomiędzy klatkami animacji

## Punktacja

- 1pkt za spełnienie wymagań definicji choinki
- 1pkt dodatkowe 'ficzery' i ogólny wygląd oceniany według subiektywnego opinii prowadzącego



Klatka 0



Klatka 1



Klatka 2