

# Kurs języka Python

## Lista 5.

### Zadanie 1.

Zaimplementuj to samo zadanie które wykonałaś/wykonałeś z listy 4., ale wykorzystaj do tego iteratory.<sup>2</sup> Zbadaj, która teraz wersja jest najszybsza.

Przetestuj działanie implementacji dla różnych argumentów, np. dla 10, 100, 1000 etc. Wypisz na konsolę czasy działania dla poszczególnych danych i implementacji w ładnie sformatowany sposób, na przykład

	imperatywna	funkcyjna	składana	iterator
10	0.002	0.001	0.002	0.003
100	0.020	0.010	0.020	0.030
1000	0.200	0.100	0.200	0.300

Można też plik sformatować tak, aby wynik mógł być wejściem dla jakiegoś pakietu/programu rysującego wykresy, np. *gnuplot* czy *matplotlib*.

### Zadanie 2.

*Kryptarytm* to zadanie, w którym litery należy zastąpić cyframi tak, aby powstało poprawne działanie. Przykładem takiego kryptarytmu jest

```

KIOTO
+ OSAKA
-----
TOKIO

```

Napisz program rozwiązujący takie kryptarytmy.

### Zadanie 3.

Poniższe zadanie polega na rekonstrukcji dwuwymiarowego obrazu na podstawie rzucanego cienia. Zakładamy, że obraz jest prostokątem czarno-białych pikseli. Cień to dwa wektory, opisujące ile jest zaczernionych pikseli w wierszu bądź kolumnie. Poniżej przykład obrazu rozmiaru  $4 \times 4$ :

2	1	3	1	
■	□	□	□	1
■	■	■	□	3
□	□	■	■	1
□	□	■	■	2

którego cień opisują dwa wektory:  $H = (2, 1, 3, 1)$ ,  $V = (1, 3, 1, 2)$ . Dla danego cienia może istnieć wiele różnych obrazów. Wystarczy, jeśli program znajdzie jeden.

### Zadanie 4.

Popularną rozrywką umysłową jest *sudoku*. Napisz program wkładający cyfry na zadaną planszę.

Na kolejną pracownię zaprogramuj zadanie 1. oraz jedno z zadań 2–4. Zadanie 1. jest warte 2 pkt., zadania 2–4: 5 pkt. Do zdobycia jest więc 7 pkt.

Uwaga: do zrobienia zadań 2–4 nie jest konieczny żaden wyrafinowany algorytm, wystarczy zwykle przeszukiwanie możliwych rozwiązań np. za pomocą pakietu *itertools*. Do każdego zadania dopisz też program który w czytelnej formie przedstawi rozwiązanie.

Marcin Młotkowski

<sup>2</sup>tj. coś co implementuje `__iter__` albo `yield`