

Zadanie 1. Napisz funkcję rozmywającą listę. Funkcja powinna przyjmować listę jako argument oraz modyfikować każdy jej element tak, aby był on średnią arytmetyczną siebie oraz swoich bezpośrednich sąsiadów. Pierwszy i ostatni element listy ma być pozostawiany bez zmian. Funkcja powinna zwrócić nową listę.

Stwórz listę 20 losowych liczb i zaprezentuj działanie funkcji. Przykładowo:

```
>>> lista
[3, 1, 2, 0, 4]
>>> blur(lista)
[3, 2.0, 1.0, 2.0, 4]
```

<https://docs.python.org/3/tutorial/datastructures.html>.

Zadanie 2. Użyj mechanizmu *list comprehensions* do wygenerowania listy 20 losowych liczb całkowitych od 0 do 10.

<https://docs.python.org/3/tutorial/datastructures.html#list-comprehensions>

Następnie wygeneruj dziesięcioelementową listę powyższych list. Dodaj dwie pierwsze wygenerowane listy do siebie element po elemencie oraz każdy z każdym. Przykład:

```
>>> a = [1, 2, 4]; b = [4, 2, -3]
>>> sumaElementPoElemencie(a, b)
[5, 4, 1]
>>> sumaKazdyZKazdym(a, b)
[5, 3, -2, 6, 4, -1, 8, 6, 1]
```

Zadanie 3. Napisz funkcję liczącą $\sin(x)$ oraz $\cos(x)$ dla zadanej wartości x , zwracającą obie te wartości naraz. Wypisz wyniki dla kątów od 0 do 90 stopni:

```
>>> sincos(0)
(0.0, 1.0)
>>> zad3()
kąt: 0; sin: 0.000; cos: 1.000
kąt: 9; sin: 0.156; cos: 0.988
[...]
kąt: 90; sin: 1.000; cos: 0.000
```

Wykorzystaj funkcję `format` do konwersji liczb zmiennoprzecinkowych na stringi reprezentujące je z dokładnością do trzech miejsc po przecinku.

<https://docs.python.org/3/tutorial/datastructures.html#tuples-and-sequences>

W jaki sposób tworzy się krotki o jednym elemencie? Jak utworzyć pustą krotkę? Czy można modyfikować krotki?

Zadanie 4. Utwórz listę o nazwie `uczelnia` zawierającą pary (`'imię i nazwisko'`, `ocena`). Przy pomocy mechanizmu *list comprehensions* wyrzuć z uczelni studentów którzy dostali ocenę 2,0 (wykorzystaj przypisanie do zmiennej).

Zadanie 5. Utwórz listę 40 liczb całkowitych od 0 do 39. Używając słowa kluczowego `del` usuń z niej co drugi element. Do powstałej tablicy wstaw -1 na początek, koniec oraz co 4 elementy. Korzystając z konstrukcji

lista[a:b] wypisz wszystkie elementy z powstałej listy po 4 elementy w linii. Usuń z listy pierwszy element o wartości -1 (funkcja remove).

Zadanie 6. Pobierz listę wbudowanych funkcji Pythona i wpisz je do listy. Odczytaj z niej nazwy funkcji zaczynających się od samogłoski i utwórz słownik łączący nazwę funkcji z jej długością. Wypisz zawartość słownika posortowaną alfabetycznie po nazwach funkcji. Jak domyślnie przeglądany jest słownik? Przykład wypisania zawartości słownika bez sortowania:

```
>>> slownik['exit']
4
>>> for nazwa, dlugosc in slownik.items():
    print(nazwa + ': ' + str(dlugosc))
input: 5
int: 3
[...]
eval: 4
```

<https://docs.python.org/3/tutorial/modules.html#the-dir-function>

Podpowiedź:

```
>>> samogloski = 'aiueoy'
>>> 'x' in samogloski
False
>>> 'e' not in samogloski
False
>>> 'e' in samogloski or 'e' not in samogloski
True
```

Zadanie 7. Utwórz moduł zawierający funkcję blur z zadania 1. oraz funkcję sincos z zadania 3. Napisz program importujący utworzony moduł i prezentujący jego działanie. W razie uruchomienia pliku modułu tak, jakby był samodzielny programem, wypisz informację co robi ten moduł.

<https://docs.python.org/3/tutorial/modules.html>

Zadanie 8. Utwórz własną funkcję imitującą działanie *list comprehensions*. Funkcja ta powinna przyjmować kolekcję elementów do przetworzenia, funkcję transformującą i funkcję określającą warunek umieszczenia w liście wynikowej.

```
>>> zad8(range(-5,5), math.fabs, lambda x: True)
[5.0, 4.0, 3.0, 2.0, 1.0, 0.0, 1.0, 2.0, 3.0, 4.0]
```

Nie używaj *list comprehensions* implementując swoją funkcję (możesz użyć na przykład funkcji insert).

<https://docs.python.org/3/tutorial/controlflow.html#lambda-expressions>

<https://docs.python.org/3/tutorial/controlflow.html#pass-statements>

<https://docs.python.org/3/tutorial/datastructures.html>

Zadanie 9. Zmodyfikuj funkcję z zadania 8 tak, aby zliczała liczbę swoich wywołań w

- 1) globalnej zmiennej,
- 2) zmiennej z zewnętrznego zakresu.