

Elementy programowania – semestr 2020/2021

Lista I

Zapoznanie z językiem Python; Deklaracja zmiennych; sekwencyjne typy danych; wyrażenia logiczne; mutable vs. immutable

Zad. 1. Użyj trybu interaktywnego jako kalkulatora, aby wyliczyć następujące wyrażenia:

- suma liczb: 2, 3, 5, -8
- średnia liczb: 2, 3, 5, -8
- $\cos(\pi)$

wskazówka:

```
```py
import math
dir(math)
help(math.cos)
```
```

- $\sin(x)$, gdzie $x = 2 / 3 * \pi$
- $\sin(x)$, gdzie $x = 2 / 3 * \pi - \pi$

!To nie jest kod tylko polecenie!

Zad. 2. Stwórz zmienną o nazwie `pi` i wartości 3.14. Wykorzystaj ją do obliczenia pola koła o promieniu 12. Wynik wyświetl na ekran. Zrób to samo importując wartość `pi` z biblioteki `math`.

Zad. 3. Korzystając z wbudowanej dokumentacji znajdź funkcję, która wyznacza licznik i mianownik ułamka zwykłego dowolnej liczby zmiennoprzecinkowej.

- Wskazówka: `help(float)*`
- Przetestuj jej działanie na kilku dowolnych przykładach.
- Przetestuj jej działanie na `math.pi`. Czy jesteś w stanie to wyjaśnić (π jest niewymierna)? Jeśli nie, to zapoznaj się z np. https://pl.wikipedia.org/wiki/Liczba_zmiennoprzecinkowa i sprawdź `sys.float_info`

Zad. 4. Zdefiniuj dwie zmienne typu tekstowego i jedną całkowitą, np.

```
```py
a = "jeden"
b = "dwa"
c = 3
```
```

Wykonaj następujące polecenia, następnie skomentuj wyniki:

```
```py
a * b
a + b
a * c
a + c
a + str(c)
```
```

Zad. 5. Korzystając z dokumentacji klasy `*string*` (``help(str)``) zapoznaj się z definicjami funkcji: ``strip()``, ``isnumeric()`` oraz ``rjust()``. Zadeemonstruj ich działanie na dowolnych przykładach.

Zad. 6. Napisz skrypt, który wykona następujące czynności:

- zmiennej ``a`` przypisze wartość podaną przez użytkownika np. ``5``
- zmiennej ``b`` przypisze wartość podaną przez użytkownika np. ``3``
- zmiennej ``P`` przypisze wartość ``a*b``
- wypisze na ekranie "Pole prostokąta o bokach ``a`` i ``b`` wynosi ``P``." (gdzie w miejsce zmiennych zostaną wstawione odpowiednie wartości).

! Wykorzystaj funkcje `input` do wykonania tego ćwiczenia ! `help(input)`
Zwróć uwagę jakiego typu zmienną zwraca funkcja `input()`

Zad. 7. Stwórz krotkę (`*tuple*`) zawierającą pięć cyfr: 0, 1, 2, 3, 4 oraz pięć literałów słownych: "pięć", "sześć", "siedem", "osiem", "dziewięć".

- Wydrukuj na ekranie trzy pierwsze elementy.
- Wydrukuj na ekranie 2 ostatnie elementy.
- Wydrukuj co drugi element (zaczynając od drugiego).
- Wydrukuj elementy krotki zaczynając od ostatniego (odwrotnie)
- Korzystając z funkcji `*len*` sprawdź ilość elementów w krotce oraz długość przedostatniego elementu.
- Niech ``x`` oznacza nazwę krotki. Wykonaj:
 - `x[:5] + (5, 6) + x[-3:]`
 - `x[:5], (5, 6), x[-3:]`

- porównaj i opisz otrzymane wyniki
- g. Dodaj pusty literał słowny na koniec krotki 'x'. Czy możesz skorzystać z funkcji `*append*`?

Zad. 8. Stwórz listę studentów: Kasia, Basia, Marek, Darek.

- a. Korzystając z metody `*append*` dodaj do listy Józka.
- b. Korzystając z metody `*extend*` dodaj do listy Anię i Basię.
- c. Posortuj alfabetycznie studentów.
- d. Wypisz na ekranie:
 - czwartego studenta na liście
 - dwóch pierwszych studentów na liście
 - dwóch ostatnich studentów na liście
- e. Usuń wszystkie Basie z listy
- f. Korzystając z funkcji `*len*` sprawdź ilość studentów.
- g. Z ostatecznej listy studentów utwórz krotkę.

Zad. 9. Korzystając z ``range`` utwórz listę zawierającą wszystkie wielokrotności liczby 3 mniejsze od 100.

- Korzystając z ``del`` usuń co trzeci element (zaczynając od piątego).
- Sprawdź definicję funkcji wbudowanej `*sum*` (``help(sum)``). Wykorzystaj ją oraz funkcję `*len*`, aby wyliczyć średnią arytmetyczną otrzymanej listy.

Zad. 10. Stwórz krotkę: ``('a', 'b', 'c', 'd')``

- Zapoznaj się z dokumentacją funkcji ``str.join``.
- Wykonaj następujące polecenia (gdzie ``x`` to zmienna wskazująca na krotkę):

```
```py
"".join(x)
" ".join(x)
", ".join(x)
```
```

- Wydrukuj na ekranie 100 zer oddzielonych tabulacjami (spróbuj wykonać to w jednej linii).

Zad. 11. Stwórz obiekt typu `*str*`, który przechowuje tekst inwokacji Pana Tadeusza:

```
```py
inwokacja = """Litwo! Ojczyzno moja! Ty jesteś jak zdrowie,
Ile cię trzeba cenić, ten tylko się dowie,
Kto cię stracił. Dziś piękność twą w całej ozdobie
Widzę i opisuję, bo tęsknię po tobie
Panno święta, co Jasnej bronisz Częstochowy
I w Ostrej świecisz Bramie! Ty, co gród zamkowy
```

Nowogródzki ochraniasz z jego wiernym ludem!  
Jak mnie dziecko do zdrowia powróciłaś cudem,  
Gdy od płaczącej matki pod Twoją opiekę  
Ofiarowany, martwą podniosłem powiekę  
I zaraz mogłem pieszo do Twych świątyń progu  
Iść za wrócone życie podziękować Bogu,  
Tak nas powrócisz cudem na Ojczyzny łono.  
Tymczasem przenoś moją duszę utęsknioną  
Do tych pagórków leśnych, do tych łąk zielonych,  
Szeroko nad błękitnym Niemnem rozciągnionych;  
Do tych pól malowanych zbożem rozmaitem,  
Wyzłacanych pszenicą, posrebrzanych żytem;  
Gdzie bursztynowy świerzop, gryka jak śnieg biała,  
Gdzie panieńskim rumieńcem dzięcielina pała,  
A wszystko przepasane jakby wstęgą, miedzą  
Zieloną, na niej z rzadka ciche grusze siedzą."""  
'''

- W interpreterze sprawdź wynik:

```
'''
>>> print(inwokacja)
>>> inwokacja
>>> inwokacja[o]
'''
```

- a. Popraw zmienną ``inwokacja``, aby tekst zaczynał się małą literą (załóżmy że nie wiesz jaka to litera).
- b. Korzystając z funkcji `*count*` sprawdź, ile razy występuje spójnik "i".
- c. Korzystając z funkcji `*count*` sprawdź, ile razy występuje litera "i" (niezależnie od wielkości).
- d. Sprawdź, czy słowo "dziecko" występuje w tekście
- e. Korzystając z funkcji `*str.split*`:
  - stwórz listę wyrazów występujących w tekście (np. 30 słów => 30 elementów)
  - stwórz listę, której każdy element odpowiada jednej linijce tekstu (4 linie => 4 elementy)
  - Popraw zmienną ``inwokacja``, tak aby każda linijka tekstu zaczynała się małą literą.

Zad. 12. Zapoznaj się z dokumentacją funkcji `*id*`.

- Wykonaj następujące polecenia:

```
``py
x = 2
y = x
print(x, y, id(x), id(y))
y = 3
```

```
print(x, y, id(x), id(y))
````
```

- Wykonaj następujące polecenia:

```
``py  
x = [1,2]  
y = x  
print(x, y, id(x), id(y))  
y[1] = 3  
print(x, y, id(x), id(y))  
````
```

- Wyjaśnij, dlaczego dlaczego modyfikacja y w drugim przypadku zmieniła także x