

Lista 1

Każde zadanie (oprócz 0) należy umieścić w osobnym pliku `imie.nazwisko.XX.YY.txt` gdzie `XX` to numer listy a `YY` numer zadania. W pliku tekstowym musi się znaleźć odpowiedni fragment kodu skopiowany z interpretera oraz odpowiedzi na zadane pytania.

Zadanie 0 (0 pkt)

Zweryfikuj instalację Pythona w swoim systemie.

```
ls -l /usr/bin/python*
```

W większości przypadków wynik powinien zawierać:

- plik wykonywalny `python2.X` dla Pythona w wersji 2.X
- plik wykonywalny `python3.X` dla Pythona w wersji 3.X
- link symboliczny `python2` do domyślnej wersji Pythona 2
- link symboliczny `python3` do domyślnej wersji Pythona 3
- link symboliczny `python` do domyślnej wersji Pythona

Upewnij się, że korzystasz z Pythona 3. Jeśli `/usr/bin/python` wskazuje na Pythona w wersji 2.X (co jest wysoce prawdopodobne), uruchamiaj Pythona poleceniem `python3`.

Jeśli `/usr/bin/python` wskazuje na `python-wrapper` możesz sprawdzić domyślną wersję korzystając z `eselect python --list`.

Zadanie 1 (1 pkt)

Wyświetl na ekranie listę funkcji wbudowanych:

```
>>> import builtins
>>> dir(builtins)
```

Następnie sprawdź dokumentację funkcji `print`:

```
>>> help(print)
```

i wykorzystaj ją, aby na ekranie wydrukować:

- tekst `Ala ma kota.`
- wynik działania `2 + 2`
- wyniki działań: `2 ** 5`, `35 // 2`, `35 / 2`, `35 % 2` oddzielone tabulacją (`\t`)

Wyjaśnij działania operatorów `**` i `%` oraz różnicę między `//` a `/`.

Zadanie 2 (1 pkt)

Utwórz zmienną `print` dowolnego typu i spróbuj wydrukować na ekranie wynik dowolnego działania matematycznego, np.

```
>>> print = "Psujemy Pythona!"
>>> print(2 + 2)
```

Wyjaśnij co się stało i dlaczego w Pythonie jest to możliwe.

Zadanie 3 (1 pkt)

Sprawdź dokumentację funkcji wbudowanej `input`. Wykorzystaj ją, aby pobrać dwie liczby całkowite i wydrukuj na ekranie ich sumę, np.

```
>>> a = input()
>>> b = input()
>>> suma = a + b
>>> print(suma)
```

Czy takiego wyniku się spodziewałaś/-eś? Wyjaśnij otrzymany rezultat.

Zadanie 4 (1 pkt)

Posiłkując się dokumentacją i stosownymi przykładami, wyjaśnij różnicę między zaokrągleniem liczby zmiennoprzecinkowej wykorzystując `int(float)` oraz `round(float)`.

Zadanie 5 (1 pkt)

W Pythonie 2 sprawdź wynik następujących działań:

```
>>> import sys      # informacje systemowe
>>> x = sys.maxint  # użyj help(sys), aby sprawdzić definicję maxint
>>> type(x)
>>> x += 1
>>> type(x)
```

Powtórz to samo w Pythonie 3. Korzystając z dowolnych źródeł (innych niż prowadzący lub koleżanki / koledzy z grupy) wyjaśnij otrzymany rezultat.

Zadanie 6 (1 pkt)

Wyjaśnij znaczenie zmiennej `_` w trybie interaktywnym i zademonstruj jej działanie na dowolnym przykładzie.

Zadanie 7 (2 pkt)

Niech $a = 3$ i $b = 4$ będą długościami boków trójkąta, a $\alpha = 47^\circ$ kątem między nimi. Wyznacz pole trójkąta. *Wskazówka: wykorzystaj implementację funkcji trygonometrycznych z biblioteki `math`.*

Zadanie 8 (2 pkt)

Korzystając z funkcji `sys.getsizeof` sprawdź, ile pamięci zajmuje:

- 0
- `2**100`
- `2**1000`

Typ logiczny (`bool`) może przyjąć jedną z dwóch wartości: `True` i `False`. Sprawdź, ile pamięci zajmują: `True` i `False`. Czy jest to wynik, którego się spodziewałaś/-eś?

Zapoznaj się z dokumentacją funkcji `isinstance`.

Wykonaj następujące polecenia:

```
>>> isinstance(0, int)
>>> isinstance(0, float)
>>> isinstance(0.0, float)
>>> isinstance(True, bool)
>>> isinstance(True, int)
```

Wyjaśnij rozmiar `True` i `False`.

Zadanie 9 (2 pkt)

Zapoznaj się z dokumentacją funkcji wbudowanej `id` (np. `help(id)`). Następnie wykonaj:

```
>>> a = 1
>>> b = 1
>>> c = 2
>>> print(id(a), id(b), id(c))
>>> b = 2
>>> print(id(a), id(b), id(c))
```

Wyjaśnij otrzymane wyniki.

Zadanie 10 (3 pkt)

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`

Wykonaj następujące obliczenia i wyjaśnij otrzymany wynik:

```
>>> x = sys.float_info.max
>>> x
>>> 2 * x
```

Wyjaśnij wynik dodawania:

```
>>> print(0.1 + 0.2)
```