

Wstęp do programowania (studia niestacjonarne)

Dr Anna Muranova

Semestr zimowy 2024/2025, UWM w Olsztynie

4. Zajęcie (Funkcje i debugger)

Ćwiczenie 1 (Debugging). Przy pomocy Debuggera wyjaśnić wartości zmiennych w każdym przebiegu pętli.

- ```
n = 5
s = 1
for i in range(1, n + 1):
 s *= i
```

```
print(s)
```

- ```
a = 1
c = 0
while (a < 3):
    for b in range(1, 3):
        c += a + b + 1
        c += 8
    a += 1
print(c)
```

Ćwiczenie 2. (a) Napisz funkcję – konwerter temperatur, która pobiera temperaturę (liczbą zmiennoprzecinkowa) i znak F albo C i zwraca temperaturę w innej jednostce. Zrób domyślną wartość jednostki 'C'.

Uruchom funkcję w programie.

(b) Napisz funkcję, która wyznacza liczbę cyfr zadanej liczby całkowitej.

Uwaga! Nie używać typ `string`.

W funkcji `main` wczytaj liczbę, a następnie wypisz wartość, zwracaną dla niej przez funkcję.

Ćwiczenie 3 (*args). • Napisz funkcję `gmean()`, która zwraca średnią geometryczną dowolnej ilości podanych liczb. W funkcje `main` zrób kilka przypadków testowych przy pomocy `assert`.

- Napisz funkcję, wyświetli dowolna ilość podanych do niej elementów w następujący sposób:

```
func(3, 'Ala', 5)
```

```
3
Ala Ala
5 5 5
```

Ćwiczenie 4. Napisz funkcję, która ma liczbą całkowitą n jako parametr i oblicza

•

$$e1 = \left(1 + \frac{1}{n}\right)^n$$

•

$$e2 = \sum_{k=0}^n \frac{1}{k!}$$

Wyświetl wyniki dla $n = 10, 100, 1000$.

Ćwiczenie 5. Napisz funkcję, która zwraca maximum

(a) z dwóch liczb

```
def my_max (a, b)
```

(b) z trzech liczb

```
def my_max (a, b, c)
```

(c) * z dowolnej ilości liczb

W funkcje `main` zrób kilka przypadków testowych kilka przypadków testowych przy pomocy `assert`.

Ćwiczenie 6. Napisz funkcję – nierekurencyjny Algorytm Euklidesa. W funkcje `main` zrób kilka przypadków testowych przy pomocy `assert`.