

Wstęp do programowania. Lista 8. Funkcje.

1. Napisz program, który dla wczytanej z klawiatury liczby a , zwróci wartości $f(a) - 3 \cdot f(a + 3) + f(-a)$, gdzie:

$$f(x) = \begin{cases} x^2 + 6x - 2, & \text{dla } x \geq -1 \\ \frac{5x - 3}{x^2 - 1}, & \text{dla } x < -1 \end{cases}$$

2. Napisz funkcję, która liczy sumę/iloczyn elementów listy. Przetestuj ją w programie zwracającym listę, której elementami są sumy wierszy w macierzy wymiaru 6×9 o losowych całkowitych elementach z przedziału $(10, 100)$.
3. Napisz funkcję, która wybiera największy element listy. Zastosuj ją w programie, który tworzy n elementową listę (n podaje użytkownik) o elementach $\sin 1, \sin 2, \dots, \sin n$.
4. Napisz funkcję, która usuwa podaną literę łańcucha znaków. Wykorzystaj tę funkcję do programu, który po wczytaniu łańcucha znaków usunie z niego litery a, o, e, i
5. Napisz funkcję, która zwraca *True* jeśli podane dwie listy mają wspólny element. Przetestuj jej działanie w programie, który porówna elementy tablicy o losowo wybranych 10-ciu elementach z zakresu $\langle -50, 100 \rangle$ z tablicą $[-2 * t * *2 - 10 * t + 111 \text{ for } t \text{ in range}(17)]$.
6. Napisz funkcję, która z danych dwóch słowników stworzy jeden słownik którego elementami są elementy z danych słowników, gdy klucz się powtarzał sumujemy wartości. Przetestuj ją na słownikach postaci $\{\text{'litera': ilość wystąpień w wyrazie}\}$, dla dwóch podanych przez użytkownika wyrazów.
7. Napisz funkcję, która dla dowolnej liczby naturalnej n zwraca liczbę naturalną k taką, że $k^2 \leq n$ oraz $(k + 1)^2 > n$. Zastosuj ją w programie, który dowolną liczbę naturalną przedstawi w postaci sumy kwadratów liczb naturalnych. Ile składników sumy dostajemy, testując program? Zmodyfikuj program, tak aby losował wielokrotnie (np. 1000 razy) liczbę naturalną i sprawdzał czy ilość elementów w jej rozkładzie na sumę kwadratów jest mniejsza równa 4. Gdy napotka rozkład dłuższy ma wyświetlić tę liczbę i rozkład.
8. Napisz funkcję, która sprawdzi czy podana liczba jest pierwsza. Następnie zastosuj ją w programie, który tworzy listę złożoną z liczb od 100 do 499, następnie usunie z niej liczby pierwsze i wyświetli nową listę.
9. Napisz funkcję, która stworzy listę b pierwszych liczb pierwszych większych bądź równych od a . Ustaw wartości domyślną $b = 10$. Wykorzystaj też w programie powyższe zadanie, przez importowanie pliku z funkcją. Przetestuj te funkcje w programie, który po wczytaniu dwóch liczb całkowitych x, y zwróci trzy listy: 10 liczb pierwszych większych równych od x , 10 liczb pierwszych większych równych od y , oraz y liczb pierwszych większych równych od $2x$.
10. Napisz funkcję, która liczy sumę dzielników podanej liczby. Zastosuj ją w programie, który ma wyświetlić wszystkie liczby doskonałe w przedziale $(1, 10000)$.
11. Napisz funkcję, która oblicza $k!$. Następnie zastosuj ją w programie, który wyznacza wybraną n -tą sumę częściową szeregu $\sum_{k=0}^{\infty} \frac{1}{k!}$. Zmodyfikuj program tak aby wyświetlił sumę pierwszych 100/1000/10000 wyrazów. Czy możemy przypuszczać, że szereg jest zbieżny, jeśli tak to do jakiej liczby?

12. Zmodyfikuj powyższy program, pisząc funkcję rekurencyjną. Przetestuj go. Czy działa tak szybko jak poprzedni?
13. Napisz funkcję, która oblicza n wyraz ciągu Fibonaciego $F(n)$. Wykorzystaj ją w programach, które wyświetlą k pierwszych elementów ciągów:

$$F(n) = \frac{1}{\sqrt{5}} \left(\frac{1 + \sqrt{5}}{2} \right)^n \quad \text{oraz} \quad \frac{F(n+1)}{F(n)} = \frac{1 + \sqrt{5}}{2}.$$

Liczbę k podaje użytkownik.

14. Napisz funkcję, która oblicza sumę cyfr w podanej liczbie. Wykorzystaj ją w programie, który oblicza takie sumy dla liczb od 1 do 100 oraz wyświetli słownik postaci $\{suma:ilość\ liczb\ o\ tej\ sumie\}$ (możesz wykorzystać funkcję z zadania 6. poprzez dołączenie pliku z tą funkcją).
15. Przeanalizuj poniższe programy. Co wyświetli Python po ich wykonaniu?

```
a) def f(a):
    x=2*a**2+7*a+8
    return x
```

```
x=1
f(x)
print(x)
```

```
b) def f(a):
    global x
    x=2*a**2+7*a+8
    return x
```

```
x=1
f(x)
print(x)
```

```
c) def f():
    x=2*a**2+7*a+8
    return x
```

```
a=1
print(f())
```

```
d) def f(a):
    b=8
    def g():
        return b**2-6*b
    return g()+a
```

```
b=4
print(f(b))
```

16. Macierz traktujemy, jako listę, której elementami są kolejne wiersze, czyli listy o elementach liczbowych tej samej długości.

- Napisz funkcję, która dla n, k naturalnych zwraca macierz zerową wymiaru $n \times k$, w przypadku braku parametru k funkcja ma zwracać macierz kwadratową $n \times n$. Przetestuj funkcję wyświetlając kilka takich macierzy.
- Napisz funkcję, która dla n naturalnego zwraca macierz identycznościową wymiaru $n \times n$. Przetestuj ją wyświetlając kilka takich macierzy.
- Napisz funkcję liczącą iloczyn dwóch macierzy. Przetestuj jej działanie na wybranych przez siebie macierzach.
- Napisz funkcję wyświetlającą macierz wierszami.

Następnie napisz program, korzystający z powyższych funkcji, który sprawdzi, które macierze z plików *macierze1.txt*, *macierze2.txt* są względem siebie odwrotne, a które po przemnożeniu dają macierz zerową.