

1024 zadania w Pythonie

Igor Nowicki

24 września 2019

Spis treści

1	Wstęp	1
2	Zadania	1
3	Bibliografia	5

1 Wstęp

Zgromadzone zadania z wielu źródeł, m.in.:

- 100+ Python challenging programming exercises
- Konkurs informatyczny Logia i Minilogia
- Olimpiada Informatyczna Gimnazjalistów oraz Olimpiada Informatyczna
- Project Euler

2 Zadania

Zadanie 1. Napisz program, który znajdzie wszystkie liczby z przedziału 2000 oraz 3200, które są podzielne przez 7, jednocześnie nie będą wielokrotnościami 5. Następnie program powinien wyświetlić sumę tych wartości na ekranie.

Zadanie 2. Stwórz funkcję `factorial(n)` która zwraca wartość silni z danej liczby naturalnej `n`. Następnie nakaż programowi wypisać na ekranie wartość silni z 8. Program powinien wyświetlić wartość 40320.

Zadanie 3. Napisz program generujący słownik zawierający wartości $(i, i*i)$ dla i z przedziału od 1 do n . Dla wartości 8 program powinien wyświetlić na ekranie następującą wiadomość:

{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64}

Zadanie 4. Napisz program który przyjmuje z konsoli ciąg liczb oddzielonych przecinkami. Program powinien wygenerować listę oraz krotkę zawierające każdą z tych liczb.

Dla danych wejściowych:

34,67,55,33,12,98

wiadomość wyjściowa powinna być następująca:

['34', '67', '55', '33', '12', '98']
('34', '67', '55', '33', '12', '98')

Zadanie 5. Stwórz klasę posiadającą przynajmniej dwie metody:

- `get_string`- metoda uzyskująca ciąg znaków ze standardowego wejścia,
- `print_string`- metoda drukująca dany ciąg znaków wielkimi literami.

Dodatkowo, przygotuj prostą funkcję testową sprawdzającą obydwie metody klasy.

Zadanie 6. Napisz program obliczający i wyświetlający na ekranie wartość wyliczaną na podstawie podanego wzoru:

$$Q = \sqrt{(2 \cdot C \cdot D)/H}$$

Poniżej podane są ustalone wartości C oraz H :

- C ma wartość 50.
- H ma wartość 30.
- D jest wartością podawaną przez użytkownika poprzez standardowe wejście.

Przykład: Załóżmy że do programu zostały podane następujące wartości:

100,150,180

Wynik wyświetlony na ekranie powinien być następujący:

18,22,24

Zadanie 7. Napisz program który przyjmuje dwie cyfry, x , y ze standardowego wejścia i tworzy dwuwymiarową tablicę. Wartość elementu w i -tym wierszu i j -tej kolumnie powinna być równa $i*j$. Uwaga: $i = 0, 1, 2, \dots, x-1, j = 0, 1, 2, \dots, y-1$,

Przykład: założmy że do programu podano następujące dane wejściowe:

3,5

Wtedy, program powinien wyświetlić następującą wiadomość:

[[0, 0, 0, 0, 0], [0, 1, 2, 3, 4], [0, 2, 4, 6, 8]]

Zadanie 8. Napisz program który przyjmuje słowa oddzielone przecinkami na standardowym wejściu i drukuje te same słowa po sortowaniu w kolejności alfabetycznej.

Przykład: założmy, że następujące dane zostały przekazane do programu:

without,hello,bag,world

Wtedy, dane wyjściowe powinny być następujące:

bag,hello,without,world

Zadanie 9. Napisz program który przyjmuje ciąg linii na wejściu i drukuje je z powrotem po zmianie wszystkich znaków na wielkie litery.

Przykład:

Hello world

Practice makes perfect

Wtedy, zwrócona wiadomość powinna być następująca:

HELLO WORLD

PRACTICE MAKES PERFECT

Zadanie 10. Napisz program który przyjmuje ciąg słów rozdzielonych spacjami na wejściu i zwraca słowa z usuniętymi powtórzeniami, posortowane alfabetycznie.

Przykład: założmy, że następujące dane wejściowe zostały dostarczone do programu:

hello world and practice makes perfect and hello world again

Wtedy, na wyjściu powinna być wyświetlona wiadomość:

again and hello makes perfect practice world

Zadanie 11. Napisz program który przyjmuje ciąg 4-cyfrowych liczb binarnych oddzielonych przecinkami, a następnie sprawdza czy są podzielne przez 5. Liczby podzielne przez 5, oddzielone przecinkami, są wyświetlane na ekranie.

Zadanie 12. Napisz program który znajdzie wszystkie liczby pomiędzy 1000 i 3000 (włączając) takie, że każda cyfra jest liczbą parzystą. Uzyskane liczby rozdzielone przecinkami powinny być wydrukowane na ekranie jako informacja zwrotna.

Zadanie 13. Stwórz funkcje `expand(s)` oraz `shorten(s)` które działają w następujący sposób:

- `expand(s)` przekształca każdy ciąg znaków postaci `'a2b3c1'` do rozszerzonej postaci: `'aabbbc'`,
- `shorten(s)` przekształca dowolny ciąg znaków do postaci skróconej. Przykład: `'aaabbccc'` jest przekształcane do postaci `'a3b2c3'`.

Zadanie 14. Write a program that accepts a sentence and calculate the number of letters and digits. Suppose the following input is supplied to the program: hello world! 123 Then, the output should be: LETTERS 10 DIGITS 3

Zadanie 15. Write a program that accepts a sentence and calculate the number of upper case letters and lower case letters. Suppose the following input is supplied to the program: Hello world! Then, the output should be: UPPER CASE 1 LOWER CASE 9

Zadanie 16. Write a program that computes the value of $a+aa+aaa+aaaa$ with a given digit as the value of a . Suppose the following input is supplied to the program: 9 Then, the output should be: 11106

Zadanie 17. Use a list comprehension to square each odd number in a list. The list is input by a sequence of comma-separated numbers. Suppose the following input is supplied to the program: 1,2,3,4,5,6,7,8,9 Then, the output should be: 1,3,5,7,9

Zadanie 18. Write a program that computes the net amount of a bank account based a transaction log from console input. The transaction log format is shown as following: D 100 W 200 D means deposit while W means withdrawal. Suppose the following input is supplied to the program: D 300 D 300 W 200 D 100 Then, the output should be: 500

Zadanie 19. A website requires the users to input username and password to register. Write a program to check the validity of password input by users. Following are the criteria for checking the password: 1. At least 1 letter between [a-z] 2. At least 1 number between [0-9] 1. At least 1 letter between [A-Z] 3. At least 1 character from [#@] 4. Minimum length of transaction password: 6 5. Maximum length of transaction password: 12 Your program should accept a sequence of comma separated passwords and will check them according to the above criteria. Passwords that match the criteria are to be printed, each separated by a comma. Example If the following passwords are given as input to the program: ABd1234@1,a F#,2w3E*,2We3345 Then, the output of the program should be: ABd1234@1

3 Bibliografia

Literatura

- [1] 100+ Python challenging programming exercises
- [2] Konkurs informatyczny LOGIA.
- [3] Olimpiada Informatyczna Gimnazjalistów
- [4] Olimpiada Informatyczna