

Zbiór zadań z Python

Spis treści

1	Podstawy	4
1.1	Zadanie – obliczanie wieku.....	4
1.2	Zadanie -kalkulator	4
1.3	Zadanie – większa z dwóch liczb	4
1.4	Zadanie – wartość bezwzględna.....	4
1.5	Zadanie – koszt biletów	5
1.6	Zadanie – czy parzysta.....	5
1.7	Zadanie – średnia z pięciu ocen.....	6
1.8	Zadanie – dzielenie 2 liczb	6
1.9	Zadanie – czy zaliczony test.....	6
1.10	Zadanie– czy liczba dodatnia, ujemna, czy 0	6
1.11	Zadanie funkcja elif – przydzielenie ocen.....	6
1.12	Zadanie elif Sprawdzamy stan konta bankowego -	7
1.13	Zadanie – gra papier, nożyce kamień	7
1.14	Zadanie – z ilu cyfr składa się liczba i jaka jest suma cyfr	7
2	Pętla For	7
2.1	Zadanie – oblicz silnię.....	7
2.2	Zadanie- wypisz liczby	7
2.3	Zadanie- wypisz liczby parzyste.....	7
2.4	Zadanie narysuj kwadrat	7
2.5	Zadanie – suma kolejnych n liczb naturalnych	7
2.6	Zadanie – średnia kolejnych n liczb naturalnych.....	8
2.7	Zadanie – wypisanie liczb w kolumnach.....	8
2.8	Zadanie – średnia i suma wczytanych n liczb	8
2.9	Zadanie- suma wczytanych n liczb	8
2.10	Zadanie – szukanie liczb z zadanego przedziału podzielnych przez k	8
2.11	Zadanie – kolejne potęgi liczby 2	8
2.12	Zadanie = – kolejne potęgi liczby 3.....	8
2.13	Zadanie – sprawdzenie poprawności wczytanej liczby	8
3	Ciągi znaków	9
4	Zadania różne	10
4.1	Zadanie – wartości funkcji liniowej	10
4.2	Zadanie – suma cyfr.....	10

4.3	Zadanie – najmniejsza z 3 liczb.....	10
4.4	Zadanie – czy wśród 3 liczb są dwie takie same?	10
4.5	Zadanie – czy liczba należy do przedziału?.....	10
4.6	Zadanie – ile wczytanych liczb?	10
4.7	Zadanie Liczby doskonałe	10
4.8	Zadanie – szukanie liczb doskonałych w zadanym przedziale.....	11
4.9	Zadanie algorytm Euklidesa.....	11
4.10	Zadanie	11
5	pętle for i while.....	12
5.1	Zadania	12
5.2	Zadanie – ile przekątnych?	13
5.3	Zadanie – sprawdzenie poprawności wczytanej liczby	13
5.4	Zadanie – sumowanie liczb aż do	13
5.5	Zadanie – średnia liczb, ile parzystych.....	13
6	funkcje	14
6.1	Zadanie – funkcja szybko.....	14
6.2	Zadanie – zamiana mil na km	14
6.3	Zadanie - zamiana Celsjusza na Kelviny.....	14
6.4	Zadanie- narysuj strzałkę.....	14
6.5	Zadanie Ciąg Fibonacciego	14
6.6	Zadanie Problem Collatza	15
6.7	Zadanie Lotto.....	15
6.8	Zadanie – zgadnij jaka to liczba	15
6.9	Zadanie – gra odgadnij słowo na podstawie anagramu	15
6.10	Zadanie haker	15
6.11	Zadanie – obliczanie pola i obwodu koła	16
6.12	Zadanie – równanie kwadratowe.....	16
7	LISTY.....	16
7.1	Zadanie – utwórz listę, dodaj element, policz.....	16
7.2	Zadanie – wypełnianie listy losowymi elementami.....	16
7.3	Utwórz listę 100 losowych liczb z zakresu <1-1000> następnie wykonaj poniższe ćwiczenia: 16	
7.4	Napisz program, który utworzy n-elementową listę składającą się z <1-x> znakowych ciągów. Wartości n oraz x podaje użytkownik. Np. n = 3, x = 5, przykładowa lista: ['thbdr','pw','oerv'], następnie wykonaj na niej następujące ćwiczenia:	17
7.5	Napisz program, który utworzy dwuwymiarową listę 4x6, gdzie każdy element będzie posiadał zawartość: *	18

7.6	Napisz program, który utworzy dwuwymiarową listę 4x6x3, gdzie każdy element będzie posiadał zawartość: *	18
7.7	Napisz program, który utworzy listę elementów będących kwadratami liczb z przedziału <2,20>	18
7.8	Napisz program, który utworzy następującą listę: [1,2,3,4,5],	18
7.9	Utwórz 2 listy 10-elementowe, gdzie wartości elementów poda użytkownik.	18
7.10	Użytkownik podaje wartość x, będącą ciągiem znaków oraz wartość y będącą liczbą naturalną.	18
7.11	Użytkownik podaje wartość trzech liczb całkowitych.	18
7.12	Utwórz listę składającą się ze wszystkich liter alfabetu.	18
7.13	Zadanie – zliczanie liter w tekście	18
7.14	Zadanie – znajdź największy i najmniejszy element listy	19
8	TABLICE	19
8.1	Zadanie - transpozycja	19
8.2	Zadanie czy symetryczna, czy diagonalna	19
9	Obsługa plików	19
9.1	Utwórz nowy plik tekstowy o nazwie test.txt,	19
10	Zadania z treścią	20
10.1	Zad. Farma	20
10.2	Zad. Skarpetki	20
10.3	Zadanie deski	21
10.4	Zad. Dwukrotność sumy cyfr	21
10.5	Zad. Najmniejsza liczba	21
10.6	Liczby pechowe	22
10.7	Fulle	22
11	Wyrażenia listowe	23
11.1	Zadanie	23
11.2	Zadanie	23
11.3	Zadanie –wypełnieni listy losowymi liczbami	23
11.4	Zadanie transparent	24
11.5	Zadanie – odczyt i zapis do pliku	24
11.6	Zadanie- Liczby binarne	24
11.7	Zadanie. Kosmiczny mecz	26
12	Tematy do powtórzenia	26

1 Podstawy

1.1 Zadanie – obliczanie wieku

Napisz program proszący użytkownika o imię i rok urodzenia, a następnie obliczający i wypisujący jego wiek.

Przykładowe wykonanie:

Podaj swoje imię:

Jasiu

Podaj rok urodzenia:

2000

Jasiu, masz 24 lat.

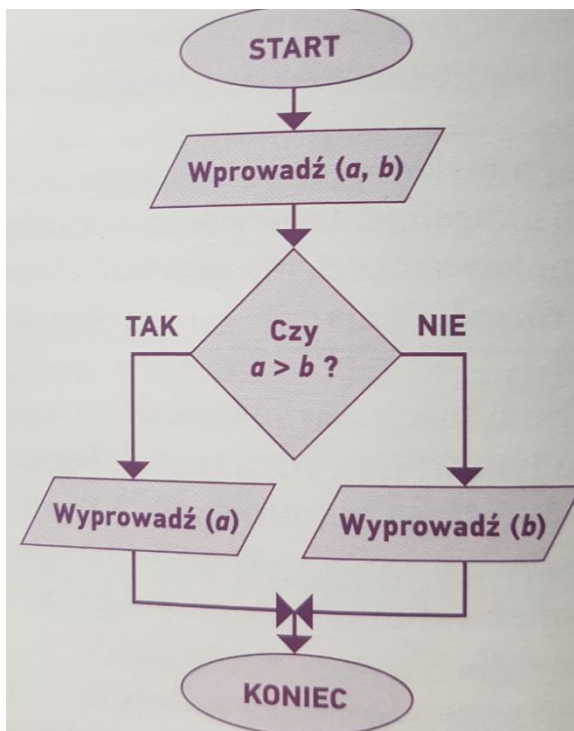
1.2 Zadanie -kalkulator

Napisz program proszący użytkownika o podanie dwóch liczb a i b i wypisujący ich sumę, różnicę, iloczyn oraz a^b i b^a .

1.3 Zadanie – większa z dwóch liczb

Na podstawie poniższego schematy blokowego napisz program w Scratch, który wybiera większą spośród dwóch podanych liczb.

Wynikiem działania programu jest informacja "większą liczbą jest" (tutaj wartość większej liczby)



1.4 Zadanie – wartość bezwzględna

Definicja

Wartością bezwzględną dowolnej liczby rzeczywistej x jest:

- ta sama liczba rzeczywista x , gdy $x \geq 0$
- liczba $-x$ (przeciwna do x), gdy $x < 0$

Matematycznie zapiszemy to tak:

$$|x| = \begin{cases} -x, & x < 0 \\ x, & x \geq 0 \end{cases}$$

Przykład: $|3|=3$; $|-5|=5$

Napisz program obliczający wartość bezwzględną podanej liczby

1.5 Zadanie – koszt biletów

Napisz program, który obliczy łączny koszt biletów, pytając:

- - ilu uczniów jedzie do kina,
- - ilu dorosłych jedzie do kina,
- - ile kosztuje bilet ulgowy (dla ucznia),
- - ile kosztuje bilet normalny (dla dorosłego),

Następnie program obliczy łączny koszt biletów i napisze:

Bilety będą kosztować (tu wstawi wyliczoną liczbę) zł.

1.6 Zadanie – czy parzysta

Napisz program, który sprawdza czy podana przez użytkownika liczba jest liczbą parzystą czy nieparzystą.

Wykorzystujemy operację modulo czyli wyznaczanie reszty z dzielenia.

Modulo – operacja wyznaczania reszty z dzielenia jednego typu liczbowego przez drugi. W dalszym ciągu napis $a \% d = r$ będzie oznaczał, iż r jest resztą z dzielenia a przez d .

Przykłady

$$12 \bmod 2 = 0$$

$$88 \bmod 2 = 0$$

$$7 \bmod 2 = 1$$

$$13 \bmod 2 = 1$$

Można zauważyć, że jeżeli reszta z dzielenia danej liczby przez 2 jest równa 0 tzn. że liczba jest parzysta. Jeżeli reszta w dzielenia jest równa 1 tzn. że liczba jest nieparzysta

1.7 Zadanie – średnia z pięciu ocen

Napisz program obliczający średnią z 5 ocen podanych przez użytkownika.

Jeżeli obliczona średnia ocen jest większa od 4,5 to wyświetlony zostanie napis: Otrzymasz nagrodę!, w przeciwnym razie napis: Nagrody nie będzie!

1.8 Zadanie – dzielenie 2 liczb

Napisz program dzielący dwie dowolne liczby.



1.9 Zadanie – czy zaliczony test

Napisz program, który zapyta na ile % napisałeś (aś) test i jeśli:

- będzie to mniej niż 30, to napisze **Nie zaliczony**
- - w przeciwnym razie napisze **Test zaliczony**.

1.10 Zadanie– czy liczba dodatnia, ujemna, czy 0

Napisz program sprawdzający czy podana przez użytkownika liczba jest dodatnia czy ujemna czy równa 0.

1.11 Zadanie funkcja elif – przydzielenie ocen

Napisz program, który na podstawie podanej przez użytkownika liczby punktów z testu, przypisze odpowiednią ocenę numeryczną:

- 90 punktów lub więcej odpowiada ocenie 5
- 80-89 punktów odpowiada ocenie 4
- 70-79 punktów odpowiada ocenie 3
- 65-69 punktów odpowiada ocenie 2

- 64 punktów lub mniej odpowiada ocenie 1

1.12 Zadanie elif Sprawdzamy stan konta bankowego -

Sprawdzamy stan konta bankowego. Użytkownik podaje nam wartość zmiennej saldo.

- Jeśli zmienna saldo jest równa 0, pojawia się na ekranie komunikat "Saldo jest równe 0, dodaj środki"
- Jeśli zmienna saldo jest liczbą dodatnią, otrzymamy wynik "Twoje saldo jest dodatnie"
- Jeśli zmienna saldo jest liczbą ujemną, wyjściem będzie komunikat "Saldo jest ujemne, dodaj środki teraz lub zostaniesz obciążony karą"

1.13 Zadanie – gra papier, nożyce kamień

Napisz grę w papier, nożyce, kamień. Użytkownik wpisuje papier, nożyce lub kamień. Następnie program losuje dla siebie papier, nożyce, kamień jako liczbę od 1 do 3 i wynik tego losowania wypisuje na konsoli. Program za pomocą warunków wypisuje kto wygrał.

1.14 Zadanie – z ilu cyfr składa się liczba i jaka jest suma cyfr

Napisz program, który prosi o podanie liczby naturalnej, a następnie wypisuje z ilu cyfr składa się wpisana wartość, a także informację o sumie tworzących ją cyfr. Dla uproszczenia załóż, że podana liczba jest poprawna (czyli rzeczywiście naturalna).

2 Pętla For

2.1 Zadanie – oblicz silnię

Napisz program obliczający silnię podanej przez użytkownika liczby naturalnej.

silnia 5 zapisujemy jako 5! $5! = 1*2*3*4*5$

2.2 Zadanie- wypisz liczby

Napisz program, który wypisuje liczby od 1 do 50.

2.3 Zadanie- wypisz liczby parzyste

Napisz program, który wypisuje liczby parzyste od 2 do 30

2.4 Zadanie narysuj kwadrat

Napisz program, który narysuje z gwiazdek (*) kwadrat 10 na 10

2.5 Zadanie – suma kolejnych n liczb naturalnych

Napisz program, który obliczy sumę n kolejnych liczb naturalnych (począwszy od 1)

2.6 Zadanie – średnia kolejnych n liczb naturalnych

Napisz program, który obliczy średnią n podawanych przez użytkownika liczb

2.7 Zadanie – wypisanie liczb w kolumnach

Napisz program, który wypisze pierwsze N liczb naturalnych w kolejności rosnącej i malejącej w osobnych kolumnach:

0	5
1	4
2	3
3	2
4	1
5	0

2.8 Zadanie – średnia i suma wczytanych n liczb

Napisz program wczytujący n liczb rzeczywistych z klawiatury. Program powinien:

- obliczyć i wypisać sumę wszystkich wczytanych liczb,
- obliczyć i wypisać sumę wczytanych liczb > 0 ,
- obliczyć i wypisać średnią arytmetyczną wszystkich wczytanych liczb,

2.9 Zadanie- suma wczytanych n liczb

Napisz program sumujący wartości ciągu n liczb podawanych przez użytkownika. Ilość liczb podaje użytkownik jako pierwszą wartość.

2.10 Zadanie – szukanie liczb z zadanego przedziału podzielnych przez k

Napisz program wyświetlający wszystkie liczby całkowite z przedziału od 50 do 100 podzielne przez dowolną liczbę k , którą podaje użytkownik. Przekształć program tak, aby przedział liczb podawał użytkownik.

2.11 Zadanie – kolejne potęgi liczby 2

Napisz program wyświetlający n kolejnych potęg liczby 2. Wartość n podaje użytkownik, musi to być liczba naturalna większa od 0.

2.12 Zadanie = – kolejne potęgi liczby 3

Napisz program wyświetlający kolejne potęgi liczby 3, aż do uzyskania potęgi większej od liczby m . Wartość m podaje użytkownik, musi to być liczba naturalna, większa od 2.

2.13 Zadanie – sprawdzenie poprawności wczytanej liczby

Napisz program pobierający od użytkownika liczbę całkowitą dodatnią. W przypadku podania liczby ujemnej lub zera, program prosi o podanie nowej wartości. Całość kończy się w momencie wprowadzenia liczby dodatniej.

3 Ciągi znaków

1. Napisz program, który wypisze długość ciągu znaków podanego przez użytkownika.
2. Napisz program, który obliczy ilość wystąpień każdego znaku w ciągu znaków podanym przez użytkownika. Następnie wypisze je bez powtórzeń na ekran.
3. Napisz program, który z ciągu znaków podanego przez użytkownika wypisze 2 pierwsze i 2 ostatnie znaki. Jeśli ciąg jest krótszy niż 2 znaki, wypisze na ekran pusty ciąg.
4. Napisz program, który zamieni w ciągu znaków podanym przez użytkownika każdy znak, który się powtórzy na @. Zmieniony ciąg znaków wypisze na ekran. Np. dla ciągu technikinformatyk wynikiem będzie technik@@forma@y@.
5. Napisz program, który wczyta od użytkownika jego imię i nazwisko. Wypisz na ekran te dane, jednak zamień ze sobą miejscami 2 pierwsze litery imienia z dwoma pierwszymi literami nazwiska.
6. Napisz program, który wczyta od użytkownika zdanie. Wszystkie tak zamień na nie i odwrotnie.
7. Napisz program, który wczyta od użytkownika zdanie. Wypisz najdłuższe słowo i jego długość.
8. Napisz program, który wczyta od użytkownika zdanie. Wypisz z niego co 3 znak.
9. Napisz program, który wczyta od użytkownika zdanie. Usuń z niego wszystkie znaki o nieparzystych indeksach. To co pozostało wypisz na ekran.
10. Napisz program, który wczyta od użytkownika zdanie. Wypisz je od końca wielkimi literami.
11. Napisz program, który wczyta od użytkownika zdanie. Wypisz to zdanie na ekran uwzględniając, że każdy jego wyraz, którego ilość znaków jest podzielna przez 3, zostaje wypisany od końca.
12. Napisz program, który wczyta od użytkownika zdanie. Wypisz z niego wszystkie litery w kolejności alfabetycznej.
13. Napisz program, który wczyta od użytkownika zdanie. Zszyfruj je szyfrem cezara.
14. Napisz program, który wczyta od użytkownika ciąg znaków. Następnie wypisze go w następujący sposób (np. technikinformatyk):
Aktualny znak t posiada indeks 0
Aktualny znak e posiada indeks 1
Aktualny znak c posiada indeks 2
...
15. Napisz program, który wczyta od użytkownika zdanie. Wypisz na ekran, jakich liter alfabetu zdanie to nie zawiera.
16. Napisz program, który wczyta od użytkownika ciąg znaków. Wypisz na ekran wszystkie niepowtarzające się litery.
17. Napisz program, który wczyta od użytkownika ciąg znaków. Wypisz na ekran pierwszy znak, który się powtórzy.
18. Napisz program, który wczyta od użytkownika ciąg znaków. Wypisz go na ekran po usunięciu spacji.
19. Napisz program, który wczyta od użytkownika ciąg znaków. Przesuń w nim wszystkie litery a na sam początek. Wypisz ciąg znaków na ekran.

20. Napisz program, który wczyta od użytkownika zdanie. Wypisz je na ekran w taki sposób, że każdy jego wyraz będzie rozpoczynał się i kończył wielką literą.
21. Napisz program, który wczyta od użytkownika zdanie. Wypisz ile jest w nim wyrazów.
22. Napisz program, który wczyta od użytkownika zdanie. Wypisz z niego najdłuższy i najkrótszy wyraz.
23. Napisz program, który wczyta od użytkownika ciąg znaków. Sprawdź czy ciąg ten jest palindromem. Np. kot – nie, kajak – tak.

4 Zadania różne

4.1 Zadanie – wartości funkcji liniowej

Napisz program obliczający wartości funkcji liniowej $ax+b$, dla kolejnych x z zakresu $\langle x_p, x_k \rangle$, gdzie a , b oraz x_p , x_k podaje użytkownik.

4.2 Zadanie – suma cyfr

Oblicz sumę cyfr liczby podanej przez użytkownika.

4.3 Zadanie – najmniejsza z 3 liczb

Napisz program wyznaczający najmniejszą z trzech liczb podanych przez użytkownika.

4.4 Zadanie – czy wśród 3 liczb są dwie takie same?

Napisz program, który odpowiada na pytanie, czy wśród trzech liczb są choć dwie takie same.

4.5 Zadanie – czy liczba należy do przedziału?

Napisz program sprawdzający czy liczba podana przez użytkownika jest z przedziału $\langle 1, 10 \rangle$ lub $\langle 17, 21 \rangle$

4.6 Zadanie – ile wczytanych liczb?

- a) Napisz program wczytujący z klawiatury liczby całkowite do momentu, gdy w ciągu wczytywanych liczb pojawi się trzecia jedynka. Program powinien zakończyć wtedy wczytywanie liczb i wypisać informację o ilości wczytanych liczb bez uwzględnienia ostatniej jedynki.
- b) Napisz program, który będzie kończył wczytywanie liczb dopiero po napotkaniu sekwencji trzech kolejno po sobie następujących jedynek?
- c) Co zrobić, aby możliwe było zakończenie wczytywania liczb po pojawieniu się żądanej sekwencji trzech różnych od siebie liczb?

4.7 Zadanie Liczby doskonałe

Liczba doskonała (definicja) to taka liczba naturalna, której suma jej dzielników właściwych (bez niej samej) jest jej równa.

Taką liczbą jest np. **6**, ponieważ

dzielniki 6 to $D_6=\{1,2,3\}$

$$6=1+2+3$$

lub **28**

dzielniki 28 to $D_{28}=\{1,2,4,7,14\}$

$$28=1+2+4+7+14$$

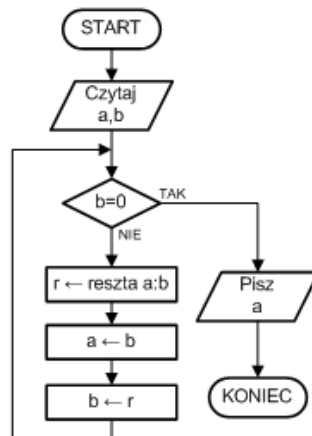
Napisz program sprawdzający czy podana przez użytkownika liczba jest liczbą doskonałą.

4.8 Zadanie – szukanie liczb doskonałych w zadanym przedziale

Napisz program znajdujący liczby doskonałe z przedziału $<2, 10000>$ Liczba doskonała to taka, której suma dzielników (mniejszych od niej samej) jest równa tej liczbie np. 6 jest liczbą doskonałą

4.9 Zadanie algorytm Euklidesa

<https://www.matemaks.pl/algorytm-euklidesa.html><https://www.matemaks.pl/algorytm-euklidesa.html>



Napisz program obliczający NWD (największy wspólny dzielnik) dwóch liczb naturalnych a i b.

Proszę o obejrzenie filmiku <https://liblink.pl/V69XEGHrbX>

4.10 Zadanie

Proszę o obejrzenie filmiku <https://liblink.pl/V69XEGHrbX>

Napisz program sortujący tablicę 10 elementową metodą bubble sort. Program nie musi być optymalny!

Przypisanie wartości obiektu a obiektowi b i wartości obiektu b obiektowi a:

Sposób 1 z wykorzystaniem zmiennej tymczasowej

```

x = 5
y = 10

# To take inputs from the user
#x = input('Enter value of x: ')
#y = input('Enter value of y: ')

# create a temporary variable and swap the values
temp = x
x = y
y = temp

print('The value of x after swapping: {}'.format(x))
print('The value of y after swapping: {}'.format(y))
  
```

Sposób 2:

```
x = 5
y = 10

x, y = y, x
print("x =", x)
print("y =", y)
```

5 pętle for i while

5.1 Zadania

- 1) Napisz program, który wypisze na ekran wszystkie liczby z przedziału <1000,5000>, są podzielne bez reszty przez 11.
 - a) podaj ile jest takich liczb w zadanym przedziale
- 2) Napisz program pozwalający użytkownikowi zgadnąć losową liczbę z przedziału <0,100>. Po każdej próbie program ma odpowiedzieć użytkownikowi, czy liczba, którą ma odgadnąć jest większa czy mniejsza od wartości strzału. W momencie trafienia program wypisuje na ekran stosowny komunikat.
- 3) Napisz program, który wczyta od użytkownika wartość liczby naturalnej n. Następnie wypisze na ekran następujący wzorec o wysokości n linii (przykład dla n = 5):

```
*
**
***
**
*
```

- 4) Napisz program, który wczytuje od użytkownika kolejne wartości liczb całkowitych do momentu, gdy ten poda wartość 0. Program wypisze wówczas na ekran komunikat ile użytkownik podał liczb parzystych, a ile nieparzystych.
- 5) Wypisz na ekran liczby z przedziału <0,10>, które nie zawierają w sobie cyfr 3 oraz 6.
- 6) Wypisz na ekran 30 pierwszych liczb ciągu fibonacciego: 0,1,1,2,3,5,8,13...
- 7) Wypisz na ekran liczby ciągu fibonacciego zawierające się w przedziale <5,100>: 5,8,13,21...
- 8) Napisz program, który przekonwertuje podaną przez użytkownika liczbę dziesiętną na liczbę binarną i wypisze ją na ekran.
- 9) Napisz program, który przekonwertuje podaną przez użytkownika liczbę binarną na liczbę dziesiętną i wypisze ją na ekran.

- 10) Napisz program, który wczyta od użytkownika wartość dwóch liczb całkowitych. Następnie obliczy i wypisze na ekran sumę wszystkich liczb całkowitych mających wartości znajdujące się pomiędzy podanymi. Np. dla 2 i 7, obliczamy sumę $3 + 4 + 5 + 6 = 18$ i wypisujemy jej wartość na ekran.
- 11) Napisz program, który wczyta od użytkownika wartość dowolnej cyfry, następnie wypisze jej tabliczkę mnożenia 1 – 10.
Np. dla 3:
 $1 \times 3 = 3$
 $2 \times 3 = 6$
 $3 \times 3 = 9$
...
 $10 \times 3 = 30$
- 12) Napisz program, który za pomocą zagnieżdżonej pętli wypisze na ekran następujący wzorzec:

```
1
22
333
4444
55555
666666
7777777
88888888
999999999
```

5.2 Zadanie – ile przekątnych?

Liczbę przekątnych n-kąta obliczamy ze wzoru

$$\frac{n * (n - 3)}{2}$$

Napisz program, który odpowie na pytanie który wielokąt ma 11780 przekątnych?

5.3 Zadanie – sprawdzenie poprawności wczytanej liczby

Napisz program pobierający od użytkownika liczbę całkowitą dodatnią. W przypadku podania liczby ujemnej lub zera, program prosi o podanie nowej wartości. Całość kończy się w momencie wprowadzenia liczby dodatniej.

5.4 Zadanie – sumowanie liczb aż do

Napisz program sumujący wartości liczb podawanych przez użytkownika. Koniec sumowania następuje wtedy, gdy użytkownik poda liczbę 0.

5.5 Zadanie – średnia liczb, ile parzystych...

Napisz program wczytujący z klawiatury liczby całkowite aż do pojawienia się pierwszej liczby <0. Program powinien obliczać i wypisywać średnią arytmetyczną wczytanych liczb > 0, iloczyn wczytanych liczb spoza przedziału [5, 100] oraz ilość wczytanych liczb parzystych.

6 funkcje

6.1 Zadanie – funkcja szybko

Samolot z Warszawy do Rzymu pokonuje dystans 1320 km. Zdefiniuj funkcję `jak_szybko(t)`, której parametrem jest czas lotu w godzinach, a wynikiem średnia prędkość samolotu.

6.2 Zadanie – zamiana mil na km

Zdefiniuj funkcję `zmiana(m)`, której parametrem jest odległość w milach lądowych, a wynikiem ta sama odległość w kilometrach.

1 mila-1,609 km

6.3 Zadanie - zamiana Celsjusza na Kelviny

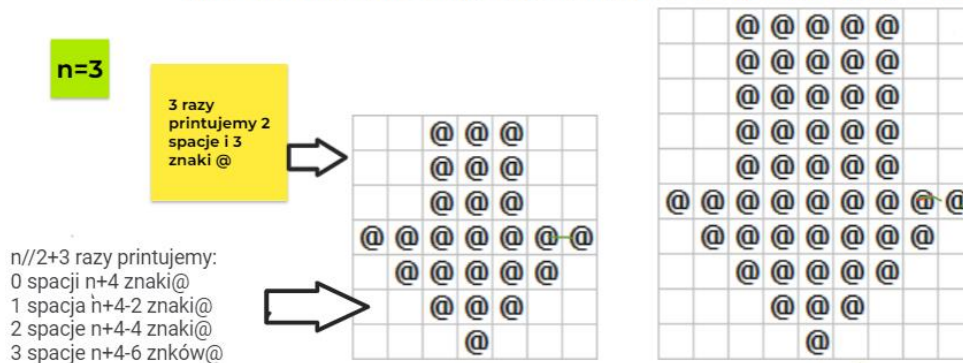
Napisz funkcję, która przyjmuje wartość temperatury w Kelvinach i zwraca wartość wyrażoną w stopniach Celsjusza: $T_C = T_K - 272.15$. W przypadku podania wartości ujemnej funkcja zwraca `None`. Przetestuj jej działanie.

6.4 Zadanie- narysuj strzałkę

vi

○ Zadanie

Napisz funkcję `strzalka(n)`, po wywołaniu której będą wypisane na ekranie znaki `@` tworzące strzałkę. Trzonek strzałki tworzy kwadrat wielkości $n \times n$, grot rozpoczyna $n+4$ znaków, każda kolejna linijka ma 2 znaki mniej, kończy się jednym znakiem. Zakładamy, że parametr jest liczbą nieparzystą z zakresu od 3 - 49.



Rysunek 1 Strzałki dla parametru 3 i 5

6.5 Zadanie Ciąg Fibonacciego

Ciąg Fibonacciego to ciąg liczb, w którym pierwszy wyraz jest równy 0, drugi jest równy 1 a każdy następny jest sumą dwóch poprzednich. Można go przedstawić za pomocą następującego zapisu:

$$F_0 = 0$$

$$F_1 = 1$$

$$F_n = (F_{n-1}) + (F_{n-2})$$

Obliczając n -ty wyraz ciągu, musisz posługiwać się wartościami poprzednimi czyli $n-1$, $n-2$ itd. aż dojdiesz do wartości które znasz. Są nimi wartości dla F_0 i F_1 .

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89

Zdefiniuj funkcję fib obliczającą n-ty wyraz ciągu Fibonacciego

6.6 Zadanie Problem Collatza

Zadanie Collatza to znane matematyczne zagadnienie, które mówi, że dla każdej dodatniej liczby całkowitej n , jeśli n jest parzyste, należy je podzielić przez 2, a jeśli n jest nieparzyste, należy pomnożyć przez 3 i dodać 1. Proces ten należy powtarzać z wynikiem, aż osiągnie się wartość 1.

Zadanie Collatza mówi, że niezależnie od wartości n , z jaką zaczniemy, ten proces zawsze osiągnie wartość 1.

6.7 Zadanie Lotto

Napisz program, w którym użytkownik podaje 6 różnych liczb z zakresu $<1,49>$, a komputer losuje 6 różnych liczb z tego zakresu. Wynikiem działania programu jest informacja ile liczb trafił użytkownik i jakie to były liczby.

6.8 Zadanie – zgadnij jaka to liczba

Napisz grę w której komputer losuje pewną liczbę w zakresie 1-1000, a użytkownik ma ją znaleźć. W tym celu wpisuje dowolne liczby i dostaje informację czy są one mniejsze czy większe od wylosowanej liczby. Po zgadnięciu jest wyświetlone za którym razem to się udało.

6.9 Zadanie – gra odgadnij słowo na podstawie anagramu

Napisz grę, w której program wyświetla anagram pewnego słowa, a zadaniem użytkownika jest odgadnięcie tego słowa. Po zgadnięciu program wyświetla za którym razem to się udało.

Przetasuj listę (zmień kolejność elementów listy):

```
import random
mylist = ["apple", "banana", "cherry"]
random.shuffle(mylist)
print(mylist)
['cherry', 'apple', 'banana']
```

6.10 Zadanie haker

Od dawna wiadomo, że hakerzy posługują się własnym językiem, różnymi skrótami i innymi formami zaciemniania, żeby zacierać ścieżki po sobie. O najlepszych z nich zwykle mawia się „h4x0rzy”. Niektórzy hakerzy czasami zamieniają niektóre litery na cyfry, żeby ich tekst wyglądał bardziej profesjonalnie. Każde wystąpienie jednej z poniższych liter w tabeli zamieniane jest na odpowiadającą jej cyfrę. Pozostałe znaki pozostają niezmiennione.

Litera	Cyfra
a	4
e	3
i	1
o	0
s	5

Napisz program, który wczyta napis, przekształci go do hakerskiego slangu zgodnie z powyższą tabelą i wypisze wynik na standardowe wyjście.

6.11 Zadanie – obliczanie pola i obwodu koła

Napisz program, który oblicza pole i obwód koła o promieniu podanym przez użytkownika. Wartość stałej π weź z biblioteki `math`. Promień nie może być ujemny. W przypadku podania liczby ujemnej, program powinien wypisywać komunikat informujący o błędnej wartości i nic nie liczyć.

6.12 Zadanie – równanie kwadratowe

Napisz program, który prosi o podanie współczynników równania kwadratowego $ax^2 + bx + c = 0$ i wypisuje rozwiązania równania (lub informację o braku rozwiązań).

7 LISTY

7.1 Zadanie – utwórz listę, dodaj element, policz

Jesteś ankietерem, Twoim zadaniem jest zebrać informację o ulubionych batonikach. Każdy ankietowany podaje swój ulubiony baton.

Zebrane dane:

"Snickers", "Bounty", "Mars", "WW", "Bounty", "Duplo", "Prince Polo", "Twix", "Snickers", "WW", "Duplo", "Bounty"

- Stwórz listę batoników z zebranych danych
- Do listy dodaj swój ulubiony baton (użyj odpowiedniej funkcji, aby dodać element na listę)
- Wyświetl całą listę
- Wyświetl pierwszy i ostatni element listy
- Policz ile osób wybrało "Bounty" jako swój ulubiony baton
- Sprawdź czy baton "BA" występuje na liście
- Usuń z listy pierwszy element (użyj odpowiedniej funkcji, aby dodać element na listę)
- Stwórz nową listę zawierającą ulubione ciastka
- Stwórz nową listę 'sweets', która będzie zawierać wszystkie batony i ciastka (połącz obie listy)

7.2 Zadanie – wypełnianie listy losowymi elementami

Wypełnij listę 10-ma liczbami losowymi od -5 do 5. Wyświetl listę, wylicz sumę elementów listy i wyświetl tę sumę.

7.3 Utwórz listę 100 losowych liczb z zakresu <1-1000> następnie wykonaj poniższe ćwiczenia:

- Napisz program, który obliczy i wypisze na ekran sumę wszystkich elementów listy.
- Napisz program, który znajdzie i wypisze na ekran najmniejszy element listy (wykonaj 2 wersje, bez użycia i używając wbudowaną funkcję języka Python).
- Napisz program, który znajdzie i wypisze na ekran największy element listy (wykonaj 2 wersje, bez użycia i używając wbudowaną funkcję języka Python).
- Napisz program, który znajdzie i wypisze na ekran medianę elementów listy.
- Napisz program, który posortuje elementy listy od najmniejszego i wypisze na ekran pierwsze 20 z nich (wykonaj 2 wersje, bez użycia i używając wbudowaną funkcję języka Python).
- Napisz program, który obliczy i wypisze na ekran iloczyn wszystkich elementów listy.

7. Napisz program, który sprawdzi i wypisze na ekran ilość liczb 3-cyfrowych na liście
8. Napisz program, który sprawdzi i wypisze na ekran liczbę oraz jej ilość powtórzeń, która najczęściej występuje na liście.
9. Napisz program, który sprawdzi i wypisze na ekran wszystkie liczby, które się nie powtarzają na liście.
10. Napisz program, który sprawdzi i wypisze na ekran, które liczby powtarzają się na liście dokładnie 3 razy.
11. Napisz program, który sprawdzi i wypisze na ekran, które liczby zawierają w sobie liczbę 21, np. 21, 213, 521.
12. Napisz program, który sprawdzi i wypisze na ekran ile liczb jest większych niż 800.
13. Napisz program, który sprawdzi i wypisze na ekran trzy najmniejsze i trzy największe liczby z listy.
14. Napisz program, który usunie z listy duplikaty, następnie wypisze na ekran ilość pozostałych elementów.
15. Napisz program, który pomiesza zawartość listy. Wypisz zawartość listy przed i po pomieszaniu jej elementów.
16. Napisz program, który przekonwertuje wszystkie elementy tablicy z int na string.
17. Wypisz wszystkie elementy listy, umieszczając przy każdym informację, ile razy występuje na liście.
18. Oblicz ile jest elementów listy, których wartość mieści się w przedziale <a,b>. Wartości a i b podaje użytkownik.
19. Oblicz ile jest elementów listy, których wartość jest parzysta.
20. Oblicz ile jest elementów listy, których wartość jest nieparzysta.
21. Napisz program, który przeniesie wszystkie elementy parzyste listy posegregowane od najmniejszego na lewą stronę listy, natomiast nieparzyste posegregowane malejąco na prawą. Np. [2,1,4,6,3,6,5] -> [2,4,6,6,5,3,1]

7.4 Napisz program, który utworzy n-elementową listę składającą się z <1-x> znakowych ciągów. Wartości n oraz x podaje użytkownik. Np. n = 3, x = 5, przykładowa lista: ['thbdr', 'pw', 'oerv'], następnie wykonaj na niej następujące ćwiczenia:

1. Napisz program, który sprawdzi i wypisze na ekran ilość znaków w liście.
2. Napisz program, który sprawdzi i wypisze na ekran ile liter 'k' zawiera lista.
3. Napisz program, który sprawdzi i wypisze na ekran ile ciągów znaków: 'kt' zawiera lista
4. Napisz program, który sprawdzi i wypisze na ekran ile ciągów znaków dłuższych niż s zawiera lista. Wartość s podaje użytkownik.

5. Napisz program, który na początku każdego elementu listy doda literę a, a na końcu literę z. Np. ['csae', 'eg', 'rr'] -> ['acsaez', 'aegz', 'arrz']
- 7.5 Napisz program, który utworzy dwuwymiarową listę 4x6, gdzie każdy element będzie posiadał zawartość: *
- 7.6 Napisz program, który utworzy dwuwymiarową listę 4x6x3, gdzie każdy element będzie posiadał zawartość: *
- 7.7 Napisz program, który utworzy listę elementów będących kwadratami liczb z przedziału <2,20>
- 7.8 Napisz program, który utworzy następującą listę: [1,2,3,4,5], następnie wypisze na ekran jej wszystkie kombinacje elementów, jednak w innym ułożeniu (permutacje). Np. [1,2,3,5,4], [5,4,3,2,1], [4,3,5,1,2], itd.
- 7.9 Utwórz 2 listy 10-elementowe, gdzie wartości elementów poda użytkownik. Utwórz 3 listę 10-elementową zawierającą różnicę elementów mających te same indeksy w 2 wypełnionych tabelach.
- 7.10 Użytkownik podaje wartość x, będącą ciągiem znaków oraz wartość y będącą liczbą naturalną.
Utwórz listę zawierającą następujący wzorec połączenia obu zmiennych: np. x = 'abc', y = 3, lista=['a1','b1','c1','a2','b2','c2','a3','b3','c3']
- 7.11 Użytkownik podaje wartość trzech liczb całkowitych.
Utwórz z nich listę. Następnie pobierze elementy listy i utwórz z nich jedną liczbę całkowitą i wypisz na ekran. Np. 23, 34, 45 -> [23, 34, 45] -> 233445
- 7.12 Utwórz listę składającą się ze wszystkich liter alfabetu.
Co n-ty element podziel listę na podlisty. Wartość n podaje użytkownik. Np. ['a','b','c','d','e','f','g','h'], n=3 -> [['a','b','c'], ['d','e','f'], ['g','h','i']]
- 7.13 Zadanie – zliczanie liter w tekście
Napisz funkcję, która dla dowolnego napisu podanego jako argument, obliczy ile razy wystąpiły w niej poszczególne litery i wypisze liczbę wystąpień każdej litery na ekran w kolejności alfabetycznej
Np. dla napisu "abzkhfhiohfnlaanfnnkdkdamd" na ekran powinno się wypisać:
- a : 4
- b : 1
- d : 2
- f : 5
- h : 2
- i : 1
- k : 3
- l : 1

```
m : 1  
n : 4  
o : 1  
z : 1
```

7.14 Zadanie – znajdź największy i najmniejszy element listy

Napisz funkcję, która dla zadanego argumentu którym jest lista, zwraca indeks i wartość najmniejszego elementu oraz indeks i wartość największego elementu. *Uwaga.* Python posiada wbudowane funkcje min, max itp. do znalezienia maksimum, minimum oraz indeksów. Można (ale nie jest to obowiązkowe) ich użyć żeby sprawdzić, czy napisana przez nas funkcja daje poprawne wyniki. Jednak w funkcji nie należy ich używać - należy przejść wszystkie elementy listy pętlą for, porównywać z czym trzeba i w ten sposób znaleźć szukane wartości.

8 TABLICE

8.1 Zadanie - transpozycja

Napisz program wypełniający tablicę kwadratową NxN ($N \leq 10$) liczbami losowymi z zakresu 0 do 10, a następnie dokonujący transpozycji tej tablicy. Tablicę przed i po transpozycji należy wypisać na ekranie.

8.2 Zadanie czy symetryczna, czy diagonalna

Napisz program wczytujący z klawiatury elementy kwadratowej tablicy NxN ($N \leq 10$) i określający, czy tablica jest:

- a) symetryczna $a_{ij} = a_{ji}$ dla $i, j = 1..N$,
- b) diagonalna: $a_{ij} = 0$ dla $i \neq j$, $i, j = 1..N$
- c) górno-trójkątna: $a_{ij} = 0$ dla $i > j$, $i, j = 1..N$.

9 Obsługa plików

9.1 Utwórz nowy plik tekstowy o nazwie test.txt,

następnie wklej do niego wybrany fragment tekstu, np. tekst Hymn Polski. Następnie wykonaj na pliku test.txt następujące ćwiczenia:

1. Odczytaj całą zawartość pliku i wypisz go na ekran.
2. Odczytaj linia po linii 8 pierwszych linii tekstu, następnie wypisz je na ekran.
3. Odczytaj linia po linii 8 ostatnich linii tekstu, następnie wypisz je na ekran.
4. Dopisz do pliku dodatkową linię tekstu z informacją o autorze dzieła.
5. Wczytaj zawartość pliku linia po linii, następnie umieść je w liście. Wypisz zawartość na ekran.
6. Wczytaj zawartość pliku linia po linii, następnie umieść je w zmiennej. Wypisz zawartość na ekran.

7. Napisz program, który odnajdzie najdłuższy wyraz w pliku.
8. Napisz program, który wypisze na ekran ilość linii pliku.
9. Napisz program, który wypisze na ekran ilość wyrazów w pliku.
10. Napisz program, który wypisze na ekran ilość znaków w pliku.
11. Napisz program, który skopiuje zawartość pliku test.txt do pliku test_kopia.txt.
12. Wypisz na ekran co drugą linię pliku test.txt.
13. Napisz program, który odczyta jedną, losową linię z pliku.
14. Napisz program, który zmieni zawartość pliku usuwając z niego przejęcia do nowej linii.

10 Zadania z treścią

10.1 Zad. Farma

Rolnik ma farmę a na niej pewną liczbę kur i krów. Razem wszystkie te zwierzęta mają dokładnie X głów oraz Y nóg. Wszystkie kury mają po dwie nogi, a wszystkie krowy mają po cztery nogi. Oczywiście zarówno kury, jak i krowy mają po jednej głowie. Ile kur oraz ile krów ma rolnik?

Napisz program, który wczyta wartości X- liczba głów oraz Y- liczba nóg i wyznaczy liczbę kur oraz liczbę krów na farmie.

Wyjście:

dwie nieujemne liczby całkowite oznaczające kolejno liczbę kur oraz liczbę krów na farmie.

Możesz założyć, że dane dobrane są tak, że rozwiązanie zawsze istnieje.

Wejście:4 10 Wyjście:3 1

Wejście: 1000 2000 Wyjście:1000 0

Wejście:1000 4000 Wyjście:0 1000

Wejście:53535353 129292928 Wyjście:42424242 11111111

10.2Zad. Skarpetki

Bajtazar, zakupił dużo pojedynczych skarpetek (białych i czarnych). Chciałby wiedzieć ile jednokolorowych (albo białych albo czarnych) par skarpetek uda mu się skompletować. Napisz program, który wczyta ciąg opisujący skarpetki Bajtazara i wyznaczy liczbę jednokolorowych par skarpetek, które można z nich skompletować.

Wejście

Niepusty ciąg liter B oraz C oznaczających kolory kolejnych skarpetek Bajtazara (B oznacza skarpetkę białą, a C skarpetkę czarną).

Wyjście

Jedna nieujemna liczba całkowita – liczba par skarpetek jednokolorowych, jakie można stworzyć ze skarpetek opisanych na wejściu.

BBCBCCBCB-4

CBCBCBCB -4

CCCBBC-2

10.3 Zadanie deski

Bajtek chce zbudować wielką, kwadratową piaskownicę. Do budowy piaskownicy są mu potrzebne zaledwie cztery deski, które muszą być równej długości. Niestety, podczas kupowania desek w tartaku Bajtek zupełnie zapomniał o tym fakcie i kupił N desek o niekoniecznie równych długościach. Bajtek może (ale nie musi) najpierw skrócić posiadane deski, a następnie wybrać cztery kawałki równej długości i zbudować z nich piaskownicę. Zauważ, że deska może jedynie być skracana, a nie dzielona, czyli nie można na przykład z jednej deski o długości 4 otrzymać dwóch desek o długości 2. Bajtek nie lubi ułamków, dlatego wszystkie długości desek są całkowite oraz wszystkie długości skróconych kawałków również muszą być całkowite.

Napisz program, który: wczyta długości desek posiadanych przez Bajtkę, wyznaczy pole największej piaskownicy, którą może zbudować i wypisze wynik na standardowe wyjście.

Wejście: jedna liczba naturalna N , określająca liczbę desek,

ciąg N liczb naturalnych $L_1, L_2, \dots, L_N$ - są to długości desek posiadanych przez Bajtkę.

Wyjście: pole powierzchni największej możliwej do uzyskania kwadratowej piaskownicy zgodnie z warunkami powyżej. Jeśli zbudowanie takiej piaskownicy nie jest możliwe, należy wypisać 0.

Przykład 1

Wejście: 7 Wejście: 6 3 7 6 5 8 10

Wyjście: 36

Przykład 2

Wejście 2 Wejście 4 8

Wyjście: 0

10.4 Zad. Dwukrotność sumy cyfr

Bajtazar wymyśla ciąg liczb naturalnych: rozpoczyna od swojej ulubionej liczby X (czyli pierwszy wyraz ciągu $A_1 = X$), natomiast każda kolejna liczba w ciągu to dwukrotność sumy cyfr poprzedniej (czyli dla każdego $i \geq 1$ zachodzi $A_{i+1} = 2 \cdot s(A_i)$, gdzie $s(k)$ oznacza sumę cyfr k).

Na przykład, jeśli $X = 1$, to pierwszymi wyrazami tego będą $A_1 = 1, A_2 = 2, A_3 = 4, A_4 = 8, A_5 = 16, A_6 = 14, \dots$

Czy potrafisz szybko wyznaczyć N -ty wyraz ciągu? Napisz program, który wczyta wartości N oraz X , po czym wyznaczy N -ty wyraz ciągu Bajtazara rozpoczynającego się od X .

Wejście dla $N=6$ $X=1$ Wyjście: 14

Wejście dla $N=4$ $X=9912$ Wyjście: 6

10.5 Zad. Najmniejsza liczba

Dane są trzy cyfry (niekoniecznie różne) - ustaw je w ciąg w taki sposób, aby stworzyć najmniejszą możliwą liczbę trzycyfrową. Liczba ta nie może mieć zer wiodących (czyli nie może zaczynać się od 0).

Wejście: trzy nieujemne liczby całkowite X, Y i Z ($0 \leq X, Y, Z \leq 9$), z których zawsze co najmniej jedna jest dodatnia.

10.6 Liczby pechowe

Liczbę naturalną nazywamy pechową, jeśli spełnia jednocześnie następujące dwa warunki:

- jej suma cyfr wynosi 13,
- zawiera co najmniej raz w swoim zapisie dziesiętnym ciąg 13 jako spójny fragment (czyli zawiera cyfry 1 i 3 stojące obok siebie). Przykładowo, liczby 139, 33133 są pechowe, natomiast 13 oraz 553 nie są liczbami pechowymi.

Napisz program, który wczyta liczbę naturalną N , wyznaczy ile jest liczb pechowych nie przekraczających N i wypisze wynik na standardowe wyjście.

Wejście: się jedna liczba naturalna N ($1 \leq N \leq 1013$).

Wyjście W pierwszym (jedynym) wierszu wyjścia powinna się znaleźć jedna liczba całkowita – liczba liczb pechowych nie przekraczających N .

Przykłady

Wejście : 1000 Wyjście: 2

Wyjaśnienie do przykładu: Pechowe liczby nie przekraczające 1000 to: 139 oraz 913.

Wejście : 123456 Wyjście: 326

10.7 Full

Bajtek uczy się gry w pokera. Bardzo wielu zasad jeszcze nie rozumie, stąd często przypadkowo tworzy swoje własne warianty tej gry. Tak jest też i w tym zadaniu, w którym rozważymy specjalny wariant wymyślony przez Bajtkę. Na potrzeby tego zadania skupimy się jedynie na figurach kart (as (A), król (K), dama (Q), walet (J), dziesiątka (T), dziewiątka (9), . . ., dwójka (2)) pomijając kolory kart (kier, karo, pik, trefl).

Full to pięć kart, z których trzy karty mają tę samą figurę oraz dwie pozostałe karty mają tę samą figurę, ale różną od tej figury, która występuje trzy razy. Na przykład układ AAKKK to full, podobnie jak TT2T2, ale już AKQJT, AAKKQ, KKKKT, czy 77789 nie są fullami.

W tym zadaniu Bajtek używa tylko jednej, standardowej talii kart, w której są po cztery karty dla każdej figury. Bajtek w swoim układzie (być może znacznie więcej niż pięciu) kart wybranych z talii poszukuje jak największej liczby fulli. Każda karta może być częścią co najwyżej jednego fulla.

Ile najwięcej fulli może znaleźć Bajtek w swoim układzie kart? Pomóż mu to ustalić.

Wejście W pierwszym (jedynym) wierszu wejścia znajduje się niepusty napis złożony ze znaków ze zbioru {A, K, Q, J, T, 9, 8, 7, 6, 5, 4, 3, 2} bez żadnych odstępów. Określa on figury kolejnych kart w układzie Bajtki. Każdy znak ze zbioru występuje w napisie co najwyżej cztery razy.

Wyjście Twój program powinien wypisać dokładnie jedną nieujemną liczbę całkowitą: liczbę fulli jakie Bajtek może znaleźć w swoim układzie.

Przykłady

Wejście: QAAAATTTKKQKQK

Wyjście : 2 Możliwe jest wybranie na przykład AATTT oraz AAKKK jako dwóch osobnych fulli (zwróć uwagę, że asy użyte do konstrukcji obu fulli to parami różne karty). Możliwe są również inne rozwiązania jak na przykład: TTAAA, QQKKK. W każdym jednak przypadku uzyskamy co najwyżej dwa fulle.

Wejście: AAKKK

Wyjście : 1 W tym przykładzie wszystkie karty tworzą fulla.

Wejście: AKQJT98765432

Wyjście: 0

11 Wyrażenia listowe

Wyrażenia listowe w Python to krótkie i eleganckie sposoby tworzenia nowych list, które opierają się na wartościach istniejących list lub innych sekwencji.

11.1 Zadanie

- Utwórz listę zawierającą kwadraty liczb od 1 do 10:
- Utwórz listę zawierającą tylko parzyste liczby z istniejącej listy:
- Utwórz listę zawierającą litery z napisu:
- Utwórz listę zawierającą słowa dłuższe niż 3 litery z istniejącej listy słów:

11.2 Zadanie

- Znajdź liczby parzyste w liście liczb od 7 do 17:
- Zamień litery w słowie na duże litery:
- Oblicz sumę kwadratów liczb od 1 do 10:
- Znajdź identyczne wartości w dwóch listach;
- Znajdź unikalne wartości w dwóch listach:
- Podaj długość najdłuższego słowa na liście słów:
- Znajdź najdłuższe słowo na liście słów
- Stwórz listę długości słów w zdaniu "The quick brown fox jumps over the lazy dog":
- Stwórz listę liczb parzystych, które są podzielne przez 4 i mniejsze od 100:
- Napisz wyrażenie listowe, które zwraca listę liczb parzystych podzielnych przez 3 z przedziału od 1 do 50.

11.3 Zadanie –wypełnieni listy losowymi liczbami

Napisz program, który:

- utworzy pustą listę i wypełni ją pięcioma liczbami losowymi z dowolnego zakresu, np. $[-1,1]$ (do generowania liczb możesz użyć np. funkcji `uniform` z modułu `random`, której wywołanie `uniform(a, b)` zwraca liczbę z przedziału $[a,b]$);
- wypisze po kolei wszystkie elementy listy wraz z indeksami w następujący sposób:
- `lista[0]: 0.1 lista[1]: -0.92` itd.
- wypisze długość listy, czyli ilość jej elementów; znajdzie i wypisze największy element listy oraz jego indeks.

11.4 Zadanie transparent

Szkoła Bajtka i Bitosi organizuje wielki festyn. W ramach przygotowań, rodzeństwo podjęło się sporządzenia jak największej liczby transparentów z napisem oij (w Bajtocji używa się wyłącznie małych liter alfabetu angielskiego).

W piwnicy swojego domu Bajtek i Bitosia znaleźli transparent zawierający długi napis w dziwnym języku. Bajtek zauważył, że można spróbować zamalować na transparencie niektóre znaki tak, aby pozostały tylko trzy literki: o, i oraz j, w tej kolejności. Bitosia jeszcze poprawiła ten plan - transparent zostanie najpierw pocięty na kilka fragmentów tak, aby z każdego z nich dało się uzyskać napis oij metodą Bajtka.

Na przykład, transparent głoszący koligacjeomijaj można podzielić na dwa takie kawałki:

koligacjeomijaj → koligacje | omijaj → oijlo ij.

Na ile najwięcej fragmentów można podzielić napis na starym transparencie, aby z każdego dało się uzyskać oij?

Wejście

W pierwszym (jedynym) wierszu wejścia znajduje się napis- ciąg małych liter alfabetu angielskiego, długości co najmniej 1 i co najwyżej 1 000 000.

Wyjście

W pierwszym (jedynym) wierszu wyjścia należy wypisać jedną liczbę całkowitą – największą liczbę fragmentów, na jakie można podzielić napis z wejścia.

Jeśli taki podział jest w ogóle niemożliwy, zamiast tego należy wypisać tylko jedno słowo NIE.

napis: oaiobjibojicj wynik 2

napis: jio wynik: NIE

11.5 Zadanie – odczyt i zapis do pliku

Napisz program, który w pętli prosi o podanie liczby naturalnej i zapisuje każdą podaną liczbę do odpowiedniego pliku: parzyste do even.txt, a nieparzyste do odd.txt.

Program kończy się, gdy wpisane zostanie 0.

Uruchomienie programu po raz kolejny powinno powodować, że nowe liczby zostaną dopisane do istniejących plików lub zostaną utworzone nowe pliki.

11.6 Zadanie- Liczby binarne

W pliku liczby.txt znajduje się 1000 liczb naturalnych zapisanych binarnie. Każda liczba zapisana jest w osobnym wierszu. Pierwsze pięć wierszy zawiera następujące liczby:

11010100111

11110111111011101

1010100111010100

1101111111111111111110101001010101001

1010110011001101010011110101010101010111

Każda liczba binarna zawiera co najwyżej 250 cyfr binarnych, co oznacza, że w wielu językach programowania wartości niektórych z tych liczb nie da się zapamiętać w pojedynczej zmiennej typu całkowitoliczbowego, np. w języku C++ w zmiennej typu int.

Napisz program, który da odpowiedzi do poniższych zadań. Odpowiedzi zapisz w pliku wynik4.txt, a każdą odpowiedź poprzedź numerem oznaczającym odpowiednie zadanie.

Zadanie 1

Podaj, ile liczb z pliku liczby.txt ma w swoim zapisie binarnym więcej zer niż jedynek.

Przykład: Dla zestawu liczb:

101011010011001100111

10001001

1000000

101010011100

100010

wynikiem jest liczba 3 (3 podkreślone liczby mają w swoim zapisie więcej zer niż jedynek).

Zadanie 2.

Podaj, ile liczb w pliku liczby.txt jest podzielnych przez 2 oraz ile liczb jest podzielnych przez 8.

Przykład: Dla zestawu liczb:

101011010011001100000 (*), (**)

10001001

100100 (*)

101010010101011011000 (*), (**)

100011

trzy liczby są podzielne przez 2 (*) i dwie liczby są podzielne przez 8 (**).

Zadanie 3

Znajdź najmniejszą i największą liczbę w pliku liczby.txt. Jako odpowiedź podaj numery wierszy, w których się one znajdują.

Przykład: Dla zestawu liczb:

101011010011001100111

10001001011101010

1001000

101010011100

1000110

najmniejsza liczba to: 1000110

największa liczba to: 101011010011001100111

Prawidłowa odpowiedź dla powyższego przykładu to: 5, 1.

11.7 Zadanie. Kosmiczny mecz

Dawno temu, w odległej galaktyce, rozegrano mecz w grę, która przypominała siatkówkę. W meczu wystąpiły dwie drużyny:

drużyna A i drużyna B. Mecz składał się z 10 000 krótkich rozgrywek. Każda rozgrywka kończyła się wygraną jednej z dwóch drużyn, za którą zwycięska drużyna otrzymywała jeden punkt. Plik mecz.txt zawiera zapis wyników kolejnych rozgrywek - jeden wiersz z napisem złożonym z 10 000 znaków A i B. Znak A oznacza, że rozgrywkę wygrała drużyna A, natomiast znak B - że rozgrywkę wygrała drużyna B.

Napisz program(-y) który(-e) znajdzie(-dą) odpowiedzi do poniższych zadań.

Do dyspozycji masz plik mecz_przyklad.txt, spełniający warunki zadania - odpowiedzi dla tego pliku podano w treściach zadań.

Możesz sprawdzać na nim działanie swojego programu.

Zadanie 1.1.

Oblicz, ile razy nastąpiła sytuacja, w której rozgrywkę wygrała inna drużyna niż rozgrywkę poprzednią (tzn. dwa kolejne znaki A lub B w opisie meczu się różnią).

Przykład: Dla napisu ABBBABA odpowiedzią jest 4.

Natomiast dla pliku mecz_przyklad.txt odpowiedzią jest 1798

Zadanie 1.2.

Pierwszy set w meczu trwa do pierwszej rozgrywki, po której któraś z drużyn ma co najmniej 1000 punktów za wygranie dotychczasowych rozgrywek, natomiast drużyna przeciwna ma co najmniej 3 punkty mniej. Drużyna, która zdobywa w secie więcej punktów od przeciwniej, wygrywa pierwszego seta.

Przykład: pierwszy set może się zakończyć wynikami: 1000:500, 997:1000, 1500:1497.

Wyniki 900:100, 999:1000, 1500:1500 nie kończą seta.

Podaj, która drużyna wygrała pierwszego seta i jaki w tym momencie był wynik (liczba zwycięskich rozgrywek drużyny A i liczba zwycięskich rozgrywek drużyny B w pierwszym secie). Dla pliku mecz_przyklad.txt odpowiedzią jest: A 1000:5

12 Tematy do powtórzenia

1. Sprawdzenie, czy podana liczba naturalna jest liczbą doskonałą.

2. Zamiana liczby z dowolnego systemu liczbowego (od dwójkowego do szesnastkowego) na system dziesiętny.
3. Zamiana liczby z systemu dziesiętnego na dowolny (od dwójkowego do szesnastkowego)
4. Obliczanie sumy elementów na przekątnej w tablicy dwuwymiarowej
5. Znajdowanie największego i i najmniejszego elementu w zbiorze (tablicy)
6. Sito Eratostenesa
7. Badanie czy podana liczba jest liczbą pierwszą czy złożoną
8. Algorytm NWD (nieoptymalny i optymalny)
9. Algorytm NWW
10. Program dodający ułamki
11. Skracanie ułamków
12. Sortowanie przez wybór
13. Sortowanie bąbelkowe
14. Sortowanie przez zliczanie
15. Przeszukiwanie binarne
16. Rekurencja
 - silnia,
 - potęgowanie,
 - ciąg Fibonacciego,
 - przeszukiwanie binarne,
 - zamiana liczby dziesiętnej na binarną,
 - algorytm Euklidesa,
 - obliczanie wartości wielomianu - schemat Hornera
17. Szyfr Cezara
18. Palindromy
19. Anagramy

Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych.

1. Zakres podstawowy. Uczeń:

1) projektuje i programuje rozwiązania problemów z różnych dziedzin, stosując: instrukcje wejścia / wyjścia, wyrażenia arytmetyczne i logiczne, instrukcje warunkowe, instrukcje iteracyjne,

funkcje z parametrami i bez parametrów, testuje poprawność programów dla różnych danych, w szczególności programuje algorytmy z punktu 1.2);

2) do realizacji rozwiązań problemów prawidłowo dobiera środowiska informatyczne, aplikacje oraz zasoby, wykorzystuje również elementy robotyki;

3) przygotowuje opracowania rozwiązań problemów, posługując się wybranymi aplikacjami: –

a) projektuje modele dwuwymiarowe i trójwymiarowe, tworzy i edytuje projekty w grafice rastrowej i wektorowej, wykorzystuje różne formaty obrazów, przekształca pliki graficzne, uwzględniając wielkość i jakość obrazów,

b) opracowuje dokumenty o różnorodnej tematyce, w tym informatycznej, i o rozbudowanej strukturze, dzieli tekst na sekcje i kolumny, tworzy spisy treści, rysunków i tabel, pracuje nad dokumentem w trybie recenzji, definiuje korespondencję seryjną,

c) gromadzi dane pochodzące z różnych źródeł w tabeli arkusza kalkulacyjnego, korzysta z różnorodnych funkcji arkusza w zależności od rodzaju danych, filtruje dane według kilku kryteriów, dobiera odpowiednie wykresy do zaprezentowania danych, analizuje dane, korzystając z dodatkowych narzędzi, w tym z tabel i wykresów przestawnych,

d) wyszukuje informacje, korzystając z bazy danych opartej na co najmniej dwóch tabelach, definiuje relacje, stosuje filtrowanie, formułuje kwerendy,

e) tworzy prezentacje, w tym z wykorzystaniem technik multimedialnych,

f) tworzy stronę internetową zgodnie ze standardami, wzbogaconą tabelami, listami, posługuje się arkuszem stylów, korzysta z oprogramowania i serwisów przeznaczonych do tworzenia stron; potrafi opublikować własną stronę w Internecie;

4) wyszukuje w sieci potrzebne informacje i zasoby, ocenia ich przydatność oraz wykorzystuje w rozwiązywanych problemach.

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

1) projektuje i tworzy programy w procesie rozwiązywania problemów, wykorzystuje w programach dobrane do algorytmów struktury danych, w tym struktury dynamiczne i korzysta z dostępnych bibliotek dla tych struktur;

2) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu, uruchamianiu i testowaniu programów;

3) przygotowując opracowania rozwiązań złożonych problemów, posługuje się wybranymi aplikacjami w stopniu zaawansowanym:

a) tworzy i edytuje dwuwymiarowe oraz trójwymiarowe wizualizacje i animacje, stosuje właściwe formaty plików graficznych,

b) stosuje zaawansowane funkcje arkusza kalkulacyjnego w zależności od rodzaju danych,

c) projektuje i tworzy relacyjną bazę złożoną z wielu tabel, formułuje kwerendy, stosuje język SQL do wyszukiwania informacji w bazie i do jej modyfikacji, uwzględnia kwestie integralności danych, bezpieczeństwa i ochrony danych w bazie;

4) współtworzy otwarte zasoby i aktywności oraz umieszcza je w sieci.

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

1) zapisuje za pomocą listy kroków lub pseudokodu i implementuje w wybranym języku programowania algorytmy poznane na wcześniejszych etapach oraz algorytmy:

- a) algorytm Euklidesa w wersji iteracyjnej i rekurencyjnej wraz z zastosowaniami,
- b) znajdowania określonego elementu w zbiorze uporządkowanym metodą binarnego wyszukiwania,
- c) generowania liczb pierwszych metodą sita Eratostenesa,
- d) jednoczesnego wyszukiwania elementu najmniejszego i największego,
- e) sortowania ciągu liczb przez scalanie,
- f) wyznaczania miejsc zerowych funkcji metodą połowienia,
- g) obliczania przybliżonej wartości pierwiastka kwadratowego,
- h) obliczania wartości wielomianu za pomocą schematu Hornera,
- i) szybkiego potęgowania liczb w wersji iteracyjnej i rekurencyjnej,
- j) rekurencyjnego tworzenia fraktali;

2) wykorzystuje znane sobie algorytmy przy rozwiązywaniu i programowaniu rozwiązań następujących problemów:

- a) rozkładania liczby na czynniki pierwsze,
- b) wykonywania działań na liczbach w systemach innych niż dziesiętny,
- c) znajdowania w ciągu podciągów o różnorodnych własnościach, np. najdłuższego spójnego podciągu niemalejącego, spójnego podciągu o największej sumie,
- d) zamiany wyrażenia na postać w odwrotnej notacji polskiej i obliczanie jego wartości na podstawie tej postaci;

3) objaśnia oraz porównuje podstawowe metody i techniki algorytmiczne oraz struktury danych, wykorzystując przykłady problemów i algorytmów, w szczególności:

- a) wyszukiwanie elementów liniowe i przez połowienie (do znajdowania elementów w zbiorze, sortowania przez wstawianie, przybliżonego rozwiązywania równań),
- b) rekurencję (do generowania ciągów liczb, potęgowania, sortowania liczb, generowania fraktali),
- c) metodę dziel i zwyciężaj (jednoczesne znajdowanie minimum i maksimum, sortowanie przez scalanie i szybkie),
- d) podejście zachłanne (do wydawania reszty, szukania najkrótszej drogi),
- e) programowanie dynamiczne (do szukania najdłuższego wspólnego podciągu),
- f) metodę szyfrowania z kluczem publicznym i jej zastosowanie w podpisie elektronicznym,
- g) struktury dynamiczne: stos, kolejka, lista (do realizacji algorytmu ONP),

h) grafy (do przedstawiania abstrakcyjnego modelu sytuacji problemowych).