

Języki skryptowe - Python

Lista 2 - listopad

Punktacja:

- 80-100: bdb
 - 70-79: db+
 - 60-69: db
 - 50-59: dst+
 - 40-49: dst
-

Zadanie 1 (2 pkt)

Napisz skrypt, który sprawdza, czy podany przez użytkownika ciąg znaków zawiera tylko unikatowe znaki (tj. każda litera / cyfra występuje nie więcej niż jeden raz).

Zadanie 2 (2 pkt)

Napisz funkcję, która w danym ciągu znaków (*string*) znajduje wszystkie cyfry i zwraca ich wszystkie permutacje.

Zadanie 3 (2 pkt)

Napisz funkcję, która z podanej listy zwraca liczbę wystąpień *stringów*, które zaczynają i kończą się tym samym znakiem.

Zadanie 4 (2 pkt)

Napisz funkcję, która zwraca prawdę, gdy podane dwie listy mają co najmniej jeden wspólny element.

Zadanie 5 (4 pkt)

Napisz program, który zwraca wszystkie *podlisty* podanej listy, np.

[1, 2, 3] -> [], [1], [1, 2], [1, 2, 3], [2], [2, 3], [3]

Zadanie 6 (4 pkt)

Napisz program, który pobiera od użytkownika liczbę całkowitą n , a następnie drukuje wszystkie liczby pierwsze mniejsze od n .

Zadanie 7 (3 pkt)

Napisz funkcję, która łączy dwa słowniki w jeden. Funkcja powinna drukować ostrzeżenie o powtarzających się kluczach.

Zadanie 8 (4 pkt)

Napisz program, który pobiera od użytkownika ciąg liczb (oddzielonych spacjami). Następnie tworzy słownik, którego kluczami są podane liczby a odpowiednimi wartościami kwadraty tych liczb. W pętli wydrukuj zawartość słownika w formacie: `klucz -> wartość`.

Zadanie 9 (3 pkt)

Niech

```
dziennik = {"Matematyka": 4, "Fizyka": 5, "Informatyka": 3}
```

Napisz skrypt, który dodaje do dziennika dwa dodatkowe przedmioty (i oceny). Następnie drukuje listę przedmiotów wraz z ocenami oraz średnią ocen.

Zadanie 10 (5 pkt)

Napisz program do tworzenia listy zakupów. Program w pętli powinien pytać użytkownika o nazwę produktu oraz jego cenę i zapisywać podane wartości w słowniku (koniec pętli po podaniu pustej nazwy produktu), np.

```
Podaj produkt: masło
Podaj cenę: 5
Podaj produkt: mleko
Podaj cenę: 3
Podaj produkt: [enter]
```

tworzy słownik `lista_zakupow = {"maslo": 5, "mleko": 3}`. Podanie produktu, który już występuje na liście, aktualizuje cenę. Na koniec program drukuje pełną listę zakupów, całkowity koszt produktów oraz najdroższy produkt (lub produkty jeśli najwyższą cenę posiada kilka produktów).

Zadanie 11 (3 pkt)

Napisz funkcję, która usuwa duplikaty (dwa różne klucze o takiej samej wartości) ze słownika.

Zadanie 12 (3 pkt)

Napisz funkcję, która zamienia wszystkie spacje na _ we wszystkich kluczach podanego słownika.

Zadanie 13 (5 pkt)

Przeanalizuj poniższy kod:

```
import random

imiona = ("Kasia", "Basia", "Marek", "Darek")
nazwiska = ("Nowak", "Burak", "Smith", "Doe")
przedmioty = ("Matematyka", "Fizyka", "Chemia")

def generuj_studenta():
    """Funkcja zwraca losowe imię i nazwisko"""
    return "{} {}".format(random.choice(imiona), random.choice(nazwiska))

def generuj_dziennik(n):
    """Funkcja generuje n studentów i przypisuje im losowe oceny"""
    studenci = []

    for i in range(n):
        # dodaje losowo wygenerowanego studenta z unikalnym id
        student = {
            "id": i,
            "student": generuj_studenta()
        }

        # generuje losowe oceny
        for przedmiot in przedmioty:
            student[przedmiot] = random.randint(2, 5)

        # dodaj studenta z ocenami do dziennika
        studenci.append(student)

    return studenci
```

```
def drukuj_dziennik(studenci):
    """Drukuje listę studentów wraz z ocenami"""
    for student in studenci:
        print("{} . {}".format(student["id"] + 1, student["student"]))

        for przedmiot in przedmioty:
            print("\t- {}: {}".format(przedmiot, student[przedmiot]))
```

-
- Dodaj funkcję, która liczy średnią ocen dla podanego studenta (tworząc nowy słownik), np.

```
{'id': 0, 'student': 'Marek Smith', 'Matematyka': 5, 'Fizyka': 2, 'Chemia': 2}
-> {'id': 0, 'student': 'Marek Smith', 'Średnia': 3}
```

Wsk. `help(dict.pop)`

- Dodaj funkcję, która liczy średnią ocen dla każdego studenta z podanej listy.
- Dodaj funkcję, która drukuje na ekranie ranking studentów (zaczynając od tych z najwyższą średnią).

Zadanie 14 (4 pkt)

Napisz program, który dla podanej liczby n (z konsoli) generuje słownik, którego kluczami są liczby całkowite od 1 do n , a wartościami listy ze wszystkimi dzielnikami danej liczby.

Zadanie 15 (3 pkt)

Napisz funkcję, który zwraca wynik mnożenia dwóch liczb bez wykorzystania operatora `*`.

Zadanie 16 (5 pkt)

Stwórz moduł `ciag_arytmetyczny.py` zawierający funkcje, które dla podanych $a1$ (pierwszy wyraz ciągu), r (różnica) oraz n zwracają:

- n -ty wyraz ciągu
 - sumę pierwszych n wyrazów ciągu
-

Napisz skrypt, który zaimportuje funkcje z `ciag_arytmetyczny.py`. Następnie:

- spyta użytkownika o $a1$, r i n
- wydrukuj na ekranie n -ty wyraz ciągu
- wydrukuj na ekranie sumę pierwszych n wyrazów ciągu

Zadanie 17 (3 pkt)

Napisz program, który dla podanego $c0$ drukuje ciąg Collatza (aż do wystąpienia liczby 1).

Zadanie 18 (5 pkt)

Stwórz moduł do konwersji między kolorami opisanymi w RGB i HEX. Następnie napisz skrypt, który demonstruje jego działanie.

Zadanie 19 (7 pkt)

Stwórz moduł do kompresji i dekompresji ciągu znaków. Moduł powinien zawierać trzy funkcje:

1. (3 pkt) `compress(string)`, która
 - zwraca `None`, gdy podany ciąg zawiera znaki inne niż litery (*wsk.* `help(str.isalpha)`)
 - lub zwraca skompresowany *string* wg reguły
`AAABBBBCAAAaDD -> A3B4CA3a2D2`
2. (3 pkt) `decompress(string)`, która odtwarza oryginalny tekst ze skompresowanego
3. (1 pkt) `test(n)`, która
 - generuje n losowych ciągów znaków (*wsk.* `string.ascii_letters`)
 - sprawdza, czy dekompresja skompresowanego ciągu znaków zwraca oryginał

Zadanie 20 (3 pkt)

Napisz program, który pobiera od użytkownika liczbę w postaci binarnej i konwertuje ją do systemu dziesiętnego. *Uwaga: program powinien zwrócić odpowiedni komunikat i przerwać pracę, gdy podane przez użytkownika dane są nieprawidłowe.*

Zadanie 21 (10 pkt)

Napisz program do nauki matematyki. Na starcie program powinien drukować ekran powitalny oraz menu, np:

```
#####  
# KURS MATEMATYKI #  
#####
```

1. Dodawanie
2. Odejmowanie
3. Mnożenie
4. Dzielenie
5. Wyjście

Wybierz działanie:

Po wyborze działania program powinien wylosować dwie liczby oraz zapytać o wynik, np

2 + 2 =

Po podaniu wyniku przez użytkownika program sprawdza, czy wynik jest poprawny oraz drukuje odpowiedni komunikat. Program działa w pętli aż zostanie wybrana opcja “Wyjście”.

Zadanie 22 (7 pkt)

Napisz program, który wymaga trzech argumentów z linii komend (wsk. `sys.argv`), czyli

```
python moj_skrypt.py arg1 arg2 arg3
```

które są długościami boków trójkąta (można założyć, że podane argumenty są liczbami). Na tej podstawie skrypt powinien wydrukować na ekranie następujące informacje:

- obwód trójkąta
- pole trójkąta
- informację czy trójkąt jest równoboczny, równoramienny czy różnoboczny
- informację czy trójkąt jest prostokątny, ostrokątny czy rozwartokątny

W przypadku złej liczby argumentów program powinien wyświetlić odpowiedni komunikat i zakończyć działanie.

Zadanie 23 (4 pkt)

Korzystając z metody równego podziału (bisekcji) znajdź przybliżone miejsce zerowe funkcji: $f(x) = x^3 + 2x^2 - 4x - 10$ w przedziale $[1;3]$.

Zadanie 24 (4 pkt)

Napisz program, który pobiera od użytkownika klucz produktu (16 cyfr), następnie:

- sprawdza poprawność wprowadzonych danych (długość 16, tylko cyfry)
- konwertuje klucz do stringa w formacie *AAAA-BBBB-CCCC-DDDD*
- sprawdza, czy podany klucz jest prawidłowy (każda z otrzymanych liczb 4-cyfrowych AAAA itd. jest podzielna przez 3)

Zadanie 25 (3 pkt)

Napisz program “*Magic 8-ball*”, który:

- prosi użytkownika o zadanie pytania typu tak/nie
- losuje jedną z 20 przygotowanych odpowiedzi
- pyta użytkownika, czy chce zadać kolejne pytanie