

Podstawy programowania (W11OPA-SI0072G)

Wstęp do programowania (W11FTE-SI0141W)

Wstęp do programowania (W11IKW-SI0080W)

## Laboratorium 5

### Listy danych

1. Napisz program, który prosi użytkownika o podanie tytułów 3 ulubionych filmów i zapamiętuje je na liście. Następnie program wypisuje elementy listy.
2. Zmodyfikuj program z zadania 1, aby wyświetlał dane w postaci tabeli:  
**tytuł: Skazani na Shawshank**  
**tytuł: Nietykalni**  
**tytuł: Zielona mila**  
...

### Przetwarzanie danych z listy

3. Napisz program, który prosi użytkownika o podanie 20 liczb. Program powinien zapamiętywać dane na liście, a następnie wyświetlać:
  - maksymalną liczbę na liście,
  - pozycję maksymalnego elementu listy (zastanów się jakie mogą być wyniki działania programu, jeśli element maksymalny nie jest unikatowy),
  - sumę liczb,
  - średnią liczb.Przygotuj dwie wersje rozwiązania
  - z wykorzystaniem wbudowanych funkcji: `max()`, `sum()` oraz metody `index()`,
  - nie wykorzystującą tych funkcji.

### Funkcje działające na listach

4. Zauważ, że wektory w przestrzeni trójwymiarowej można reprezentować jako listy trójelementowe. Napisz program, który prosi użytkownika o podanie współrzędnych dwóch wektorów, a następnie oblicza ich iloczyn skalarny

$$\vec{a} \circ \vec{b} = a_x \cdot b_x + a_y \cdot b_y + a_z \cdot b_z.$$

5. Rozszerz możliwości programu o funkcję, która oblicza cosinus kąta między dwoma wektorami zadawanymi jako trójelementowe listy liczb. Wykorzystaj w implementacji zależność

$$\vec{a} \circ \vec{b} = a \cdot b \cdot \cos \phi,$$

gdzie symbole  $a$  i  $b$  oznaczają długości wektorów  $\vec{a}$  i  $\vec{b}$ .

Podstawy programowania (W11OPA-SI0072G)  
Wstęp do programowania (W11FTE-SI0141W)  
Wstęp do programowania (W11IKW-SI0080W)

Przed implementacją funkcji obliczającej cosinus kąta, zaimplementuj funkcję obliczającą długość wektora

$$a = \sqrt{\vec{a} \circ \vec{a}}.$$

6. Rozszerz możliwości programu o obliczanie iloczynu wektorowego. Zaimplementuj odpowiednią funkcję, zademonstruj jej działanie.
7. Zaimplementuj funkcję, która usuwa z listy wszystkie powtórzenia. Po wywołaniu funkcji na liście powinny pozostać elementy unikatowe. Funkcja ma zachowywać kolejność pierwszego wystąpienia każdego elementu.

#### *Wycinki*

8. Napisz program, który generuje iteracyjnie listę zawierającą elementy ciągu Fibonacciego, a następnie wypisuje co drugi element tej listy. W tym celu wykorzystaj wycinek listy.

#### *Krotki*

9. Zaimplementuj funkcję, która analizuje liczby zebrane na liście. Funkcja powinna przyjmować jeden argument (listę liczb), natomiast powinna zwracać krotkę zawierającą: liczbę elementów, średnią, estymator odchylenia standardowego. Zademonstruj działanie funkcji w programie, wykorzystaj rozpakowywanie krotki.

Karol Tarnowski  
Wrocław, 2024