



Zoznamy

Kryštof Mrózek (445429@mail.muni.cz)

Kristína Tomanková (kristinatomankova@mail.muni.cz)

Radoslav Brunovský (rbrunovsky@mail.muni.cz)

Čo je to zoznam?

- Zoznam (po angl. *List*) je hodnota, ktorá obsahuje viacero hodnôt ktoré sú **zoradené**.
- Hodnoty uložené v zozname sa nazývajú *prvky* (po angl. *elements/items*)
- Prvky zoznamu môžu byť hocijaké iné dátové typy (čísla, stringy, pravdivostné hodnoty...) alebo aj ďalšie zoznamy.
- K prvkom zoznamu máme prístup a môžeme s nimi robiť rôzne operácie

```
1 zoznam = [1, 5.1e23, "python", "", True, [1, 2]]
```

Práca s prvkami zoznamu

- S prvkami zoznamov pracujeme pomocou *indexov*
- Indexy sú **celé čísla - integer**
- **Pozor!** Prvý prvok zoznamu má index **0**
- Na prvky zoznamu môžeme aplikovať operátory

```
1 zoznam = ["macka", "pes", "mys", "krava"]
2
3 print(zoznam[0])
4 print(zoznam[3])
```

Cvičenie

```
1  ciska = [1, 2.5, 10, 58]  
2  
3  print(ciska[0]+ciska[1])  
4  print(ciska[2]/ciska[3])  
5  print(ciska[3]*ciska[4])
```

Cvičenie - negatívne indexovanie

```
1  cisla = [1, 2.5, 10, 58, 55.7, 2e-3]
2
3  print(cisla[-1])
4  print(cisla[-3])
```

Slice

- Spôsob ako vytiahnuť „podzoznam” zo zoznamu

```
cisla[prvok_ktorym_zacinam:prvok_na_ktorom_koncim:krok]
```

Cvičenie

```
1  ciska = [1, 2.5, 10, 58, 55.7, 2e-3]
2
3  print(ciska[2:4])
4  print(ciska[3:])
5  print(ciska[:4])
6  print(ciska[::2])
```

Užitečná funkcia - *len()*

- Funkcia *len()* vám vráti dĺžku zoznamu

```
cisla = [1, 2.5, 10, 58, 55.7, 2e-3]
print(len(cisla))
```

[1] ✓ 0.0s

... 6

Spájanie a replikovanie zoznamov

- Zoznamy môžeme spájať pomocou operátora +
- Replikovať ich môžeme pomocou operátora *

```
cisla = [1, 2.5, 10, 58, 55.7, 2e-3]
print(cisla + [8, 9])
```

```
[2] ✓ 0.0s
```

```
... [1, 2.5, 10, 58, 55.7, 0.002, 8, 9]
```

```
cisla = [1, 2.5, 10, 58, 55.7, 2e-3]
print(cisla*2)
```

```
[3] ✓ 0.0s
```

```
... [1, 2.5, 10, 58, 55.7, 0.002, 1, 2.5, 10, 58, 55.7, 0.002]
```

Odstránenie hodnoty zo zoznamu a zmena hodnoty v zozname

```
▷ ▾  
    ciska = [1, 2.5, 10, 58, 55.7, 2e-3]  
    del ciska[2]  
    print(ciska)  
[4] ✓ 0.0s  
... [1, 2.5, 58, 55.7, 0.002]
```

```
    ciska = [1, 2.5, 10, 58, 55.7, 2e-3]  
    ciska[4]=99  
    print(ciska)  
[5] ✓ 0.0s  
... [1, 2.5, 10, 58, 99, 0.002]
```

Operátor *in*

- Operátor *in* zistí či je hodnota v zozname a vráti boolean hodnotu

```
cisla = [1, 2.5, 10, 58, 55.7, 2e-3]
print(2.5 in cisla)
print(50 in cisla)
print(50 not in cisla)
print(2.5 not in cisla)
```

✓ 0.0s

Metódy

- Metódy sú niečo ako funkcie (presnejšie vysvetlenie bude v neskorších prednáškach)
- Volajú sa „na” premenných
- Pre zoznamy máme metódy napríklad: *index()*, *append()*, *insert()*, *remove()*, *sort()*
- Tieto metódy urobia niečo so zoznamom na ktorých ich zavoláte

index()

```
cisla = [1, 2.5, 10, 58, 55.7, 2e-3]  
print(cisla.index(55.7))
```

✓ 0.0s

4

append()

```
cisla = [1, 2.5, 10, 58, 55.7, 2e-3]  
cisla.append(108)  
print(cisla)
```

```
[12] ✓ 0.0s
```

```
... [1, 2.5, 10, 58, 55.7, 0.002, 108]
```

insert()

```
cisla = [1, 2.5, 10, 58, 55.7, 2e-3]  
cisla.insert(3, "string")  
print(cisla)
```

[14] ✓ 0.0s

... [1, 2.5, 10, 'string', 58, 55.7, 0.002]

remove()

```
▶ v
    ciska = [1, 2.5, 10, 58, 55.7, 2e-3]
    ● ciska.remove(2.5)
    print(ciska)

[15] ✓ 0.0s

... [1, 10, 58, 55.7, 0.002]
```


sort()

```
cisla = [2.5, 1.0, 10, 100, 58, 55.7, 2e-3]  
cisla.sort()  
print(cisla)
```


```
[16] ✓ 0.0s
```

```
... [0.002, 1.0, 2.5, 10, 55.7, 58, 100]
```

List-like typy

- N-tice (*tuples*) a stringy sú podobné typy ako zoznamy, ale sú nemenné
- Niekoľko metód a vecí čo sme si ukazovali fungujú aj na nich (indexovanie, *len()*, *in*)
- Nemôžeme ale meniť ich prvky

```
ntica = (0, 1, 2)
print(ntica[1])
ntica[1] = 3
```

[17]  0.5s

... 1

... -----

TypeError Traceback (most recent call last)

Cell In[17], line 3

```
 1 ntica = (0, 1, 2)
 2 print(ntica[1])
----> 3 ntica[1] = 3
```

TypeError: 'tuple' object does not support item assignment

Záverečné cvičenie

Naprogramujte funkciu ktorá vezme zoznam, odstráni z neho prvý a posledný prvok a vráti takto upravený zoznam.

Záverečné cvičenie

Vytvorte funkciu, ktorá vezme dva 3D vektory a vypíše ich súčet a ich skalárny súčin.