Programování 1 (NPRG030)

Cvičení 2: Řetězec (String) a Seznam (List)

Shrnutí přednášky: Řetězec

```
# Vytvoření řetězce (angl. string)
retezec = "libovolný obsah" #  Vytvoření řetězce
```

```
# Základní operace s řetězci
len(retezec)           # Určení délky řetězce
retezec1 + retezec2    # Spojení dvou řetězců
kolikrat * retezec     # Opakování řetězce
retezec[odkud:kam]     # Vyjmutí podřetězce z řetězce (slicing)
podretezec in retezec # Test, zda je podřetězec obsažen v řetězci (vrací bool)
podretezec not in retezec # Test, zda není podřetězec obsažen v řetězci (vrací bool)
```

```
# Vybrané metody řetězce
retezec.upper()        # Převede znaky řetězce na velká písmena
retezec.lower()        # Převede znaky řetězce na malá písmena
retezec.strip()        # Odstraní tzv. whitespaces ze začátku (lstrip) i konce řetězce (rstrip)
retezec.replace(podretezec1, podretezec2) # Nahrazení prvního podřetězce za jiný v řetězci
retezec.format(objekt...) # Formátování řetězce
retezec.count(podretezec) # Vrátí počet výskytů podřetězce v řetězci
retezec.index(podretezec) # Vrátí index prvního výskytu podřetězce nebo vyhodí ValueError
retezec.index(podretezec, pozice) # Hledá výskyt až od určené pozice v řetězci
retezec.split()        # Rozdělí řetězec na seznam podle mezer (nebo zadaného separátoru)
retezec.join(seznam)  # Spojí prvky seznamu do řetězce s použitím zadaného oddělovače
retezec.startswith(podretezec) # Test, zda řetězec začíná podřetězcem podretezec (vrací bool)
retezec.endswith(podretezec) # Test, zda řetězec končí podřetězcem podretezec (vrací bool)
```

Shrnutí přednášky: Seznam

```
# Vytvoření seznamu (angl. list)
seznam = [hodnota1, hodnota2, ...] # vytvoření seznamu z hodnot
```

```
# Základní operace se seznamy
len(seznam)             # Určení délky seznamu
seznam1 + seznam2       # Provede zřetězení (konkatenace) seznamů
seznam * kolikrat       # Zopakování seznamu několikrát
seznam[index], seznam[odkud:kam] # Přístup k prvkům seznamu podle indexu
del seznam[index]; del seznam[odkud:kam] # Smazání prvku (nebo části) ze seznamu
prvek in seznam         # Test, zda je prvek obsažen v seznamu (vrací bool)
prvek not in seznam     # Test, zda prvek není obsažen v seznamu (vrací bool)
if seznam:              # Test, zda je seznam prázdný (vrací bool)
```

```
# Metody seznamu
seznam.append(prvek)  # Přidá prvek na konec seznamu
seznam.extend(seznam) # Přidá prvky z druhého seznamu na konec prvního
seznam.pop()          # Odstraní a vrátí poslední prvek seznamu (nebo prvek na určené pozici)
seznam.remove(prvek)  # Odstraní první výskyt zadaného prvku
seznam.clear()        # Vymaže všechny prvky ze seznamu
seznam.sort()         # Seřadí prvky v seznamu vzestupně
seznam.reverse()      # Obrátí pořadí prvků v seznamu
seznam.index(prvek)   # Vrátí pozici prvního výskytu zadaného prvku
```

dokumentace: https://docs.python.org/3/library/stdtypes.html#lists

Příklad 2.4: Formátování řetězce

- Napište *program* v jazyce Python, který bude *přijímat informace od účastníků* konference a *generovat potvrzení registrace* v následujícím formátu:

```
Potvrzení registrace:
Jméno: [Jméno účastníka]
Příjmení: [Příjmení účastníka]
E-mail: [E-mail účastníka]
Telefon: [Telefonní číslo účastníka]
```

- Program by měl nejprve přijmout vstupní údaje od uživatele a poté použít formátování řetězců k vytvoření potvrzení registrace
  - Zkuste použít dva různé způsoby formátování řetězce

Cvičení 2.5: Zřetězení textu

- Vytvořte *program* v jazyce Python, který načte následující text po řádcích:

```
dva DOMY, oba STEJNĚ urozené
V krásné VERONĚ, kam NÁŠ děj NÁS táhne
ZÁŠŤ dávnoletá V nový ROZBROJ žene
V němž RODNÁ ruka RODNOU krví VLÁHNE
```

- Následně *jednotlivé řádky zřetězte* (spojte) do jednoho řetězce
  - Pro zřetězení použijte vhodný separátor (tj. oddělovač)
- Zřetězený text *vypište na standardní výstup*

Cvičení 2.6: Oprava gramatiky editací řetězce

- Uložte následující text jako řetězec v jazyce Python

```
dva DOMY, oba STEJNĚ urozené
V krásné VERONĚ, kam NÁŠ děj NÁS táhne
ZÁŠŤ dávnoletá V nový ROZBROJ žene
V němž RODNÁ ruka RODNOU krví VLÁHNE
```

- Pomocí vhodných metod *doplňte interpunkci*
  - Tj. přidejte chybějící čárky a tečku na konci věty
- Opravte *velikost písmen* tak, aby odpovídala české gramatice.
  - Tj. velké písmeno na začátku věty a název místa
- Opravený text *vypište na standardní výstup*

Cvičení 2.7: Hledání v textu

- Vytvořte *program* v jazyce Python, který s textem napravo:
  - Určí délku textu (tj. *počet znaků*)
  - Nalezněte *index prvního* a *druhého výskytu* libovolně vybraného *podřetězce*
    - Lze vyhledat najednou i více než jediné slovo? Např. celou větu?
- Vše *vypište na standardní výstup*

- **Těžší varianta**
  - Ošetřete případy, kdy se hledané slovo v textu nenachází nebo nachází právě jednou

```
Dva domy, oba stejně urozené,
v krásné Veroně, kam náš děj nás táhne,
zášť dávnoletá v nový rozbroj žene,
v němž rodná ruka rodnou krví vláhne.
Z osudných ledví nepřátel těch berou
pod přenešťastnou hvězdou žití svoje
dva milující, kteří sudbou šerou.
v své smrti pohřbí roditelův boje.
Děj strašný lásky smrtí znamenané,
hněv rodičů, z něhož se dál hněv prýští,
ať teprv zmarem dítek poustane,
dvě hodiny teď půjdou po jevišti.
A budete-li trpělivost míti,
co vadno zde, chcem snahou nahraditi.
```

Cvičení 2.8: Seznam

- Vytvořte *program* v jazyce Python, který přečte text napravo a *vypíše počet slov* v textu na standardní výstup

- Nápověda:
  - Můžete převést text na seznam slov
    - Použijte `list`
  - Nezapomeňte odstranit veškerou interpunkci

```
Dva domy, oba stejně urozené,
v krásné Veroně, kam náš děj nás táhne,
zášť dávnoletá v nový rozbroj žene,
v němž rodná ruka rodnou krví vláhne.
Z osudných ledví nepřátel těch berou
pod přenešťastnou hvězdou žití svoje
dva milující, kteří sudbou šerou.
v své smrti pohřbí roditelův boje.
Děj strašný lásky smrtí znamenané,
hněv rodičů, z něhož se dál hněv prýští,
ať teprv zmarem dítek poustane,
dvě hodiny teď půjdou po jevišti.
A budete-li trpělivost míti,
co vadno zde, chcem snahou nahraditi.
```

Cvičení 2.9: Indexování seznamu

- Rozšiřte program ze cvičení 2.8 tak, že navíc na standardní výstup vypíšete:
  - *První a poslední* slovo
  - *Sedmé* slovo
  - *Třetí* slovo *od konce*

- **Těžší varianta**
  - Zkuste vypsat *sté slovo* v listu nebo devadesáté slovo indexované od konce listu
    - *Ošetřete program*, aby v případě *přetečení indexu* listu vypsal chybové hlášení

Cvičení 2.10: Editace seznamu

- Rozšiřte program ze cvičení 2.9 tak, že:
  - *Vložíte* libovolné slovo na *začátek* seznamu
  - Vložíte libovolné slovo na *konec* seznamu
  - Vložíte libovolné slovo *na sedmou pozici* v seznamu
  - *Odstraníte* ze seznamu *třetí* slovo *od konce*
- Po provedení všech úprav vypište obsah listu na standardní výstup

Cvičení 2.11: Uspořádání slov podle abecedy

- Upravte program z cvičení 2.10 následovně:
  - *Seřaďte slova* v listu podle abecedy
  - Seřazený seznam slov *vypište na standardní výstup*

- **Těžší varianta**
- Nejprve *odstraňte opakující se slova* v seznamu a teprve poté proveďte jejich uspořádání podle abecedy

Cvičení 2.12: Spojení dvou listů

- Napište program v jazyku Python, ve kterém vytvořte dva listy s libovolným obsahem a *spojte* tyto *dva listy* v jediný list
  - V prvním listu můžete uvést např. jména osob a ve druhém listu jména domácích mazlíčků
- Obsah spojeného listu vypište na standardní výstup

- Otázky:
  - V jakém pořadí budou elementy původních listů uloženy v novém, tj. spojeném, listu?
  - Pokud oba původní listy obsahovaly stejnou hodnotu, bude se ve spojeném listu hodnota vyskytovat pouze jednou nebo vícekrát?

Reference

Řetězec (angl. string)

- https://www.w3schools.com/python/python_strings.asp
- https://www.w3schools.com/python/python_strings_methods.asp
- https://www.w3schools.com/python/python_strings_format.asp
- https://www.w3schools.com/python/python_string_formatting.asp
- https://www.w3schools.com/python/python_strings_escape.asp

Seznam (angl. list)

- https://www.w3schools.com/python/python_lists.asp
- https://www.w3schools.com/python/python_lists_sort.asp
- https://www.w3schools.com/python/python_lists_copy.asp
- https://www.w3schools.com/python/python_lists_join.asp
- https://www.w3schools.com/python/python_lists_methods.asp