Domácí úkoly 8

Na začátek trocha práce se seznamy. Některé z úkolů – a zvlášť ten poslední – potřebují trochu přemýšlení; nebudeš-li vědět jak dál, zeptej se na Internetu nebo na sraze ostatních, a řešte společně!

- o. Udělej si seznam domácích zvířat. Budeš ho potřebovat v dalších úlohách. Domácí zvířata jsou např.: "pes", "kočka", "králík", "had".
- 1. Napiš funkci, která vypíše jména domácích zvířat, která jsou kratší než 5 písmen.
- 2. Napiš funkci, která vypíše jména domácích zvířat, která začínají na k.
- 3. Napiš funkci, která dostane slovo a zjistí, jestli je v seznamu domácích zvířat.
- 4. Napiš funkci, která dostane dva seznamy jmen zvířat, a vrátí tři seznamy:
 - (a) Zvířata, která jsou v obou seznamech
 - (b) Zvířata, která jsou v jen prvním seznamu
 - (c) Zvířata, která jsou v jen druhém seznamu

Napiš (a pusť) k této funkci testy, aby sis ověřila že funguje správně.

- 5. Napiš program, který seřadí seznam domácích zvířat podle abecedy.
- 6. Had byl pyšný na to, že je v abecedě první. Dokud nepřiletěla "andulka". Abys hada uklidnila, vytvoř funkci, která zvířata seřadí podle abecedy, ale bude ignorovat první písmeno (t.j. vrátí ["had", "pes", "andulka", "kočka", "králík"]).
 Postup:
 - Máš seznam hodnot, které chceš seřadit podle nějakého klíče. Klíč se dá z každé hodnoty vypočítat.
 - Vytvoř seznam dvojic [klíč, hodnota].
 - Seřaď tento seznam dvojic dvojice se řadí nejdřív podle prvního prvku, pak druhého, atd.
 - Nakonec vytvoř ze seznamu dvojic opět jen seznam hodnot.

Proč má zrovna had takovéhle výsadní postavení, zjistíš později.

Jedna klasická programovací úloha.

7. Napiš funkci, která převede římské číslice na číslo (int). Nápověda: Nejdřív napiš k této funkci nějaké testy, aby sis ověřila že (a co) (ne)funguje správně.

Dadaistický koutek: procvičení práce se seznamy, řetězci, a soubory. Tyhle úkoly opět vyžadují přemýšlení.

- 8. Vyber si básničku, která má aspoň tři sloky po aspoň třech verších. Ulož ji do souboru basnicka.txt,
- 9. Napiš program, který vypíše básničku ze souboru basnicka.txt, ale obrátí pořadí veršů (t.j. jako první vypíše poslední řádek, atd.)
 Nápověda: Každý seznam má metodu reverse, která ho "obrátí".
- 10. Napiš program, který obrátí pořadí slov v jednotlivých verších.
- 11. Obrať pořadí slok (ty by měly být oddělené jedním prázdným řádkem).
- 12. Vypiš slova básně v náhodném pořadí.
 Bonusový úkol: Snaž se přitom co nejlépe zachovat strukturu básně (sloky, verše, interpunkci, velká písmena, ...)

Ükoly 13-19 závisí jeden na druhém, řeš je postupně. Až to uděláš, můžeš si zahrát hru! Tahle sekce není jednoduchá (a poslední dva úkoly jsou obzvláště náročné), ale můžeš zkusit spojit síly s ostatními.

Jak na to? Udělej tabulku (seznam seznamů) se samými tečkami, a pak na příslušných místech nahraď tečky X-ky. Tabulku pak vypiš.

14. Napiš funkci pohyb, která dostane seznam souřadnic a světovou stranu ("s", "j", "v" nebo "z"), a přidá k seznamu poslední bod "posunutý" v daném směru. Např.:

```
souradnice = [(0, 0)]
pohyb(souradnice, 'v')
print(souradnice)  # → [(0, 0), (0, 1)]
pohyb(souradnice, 'v')
print(souradnice)  # → [(0, 0), (0, 1), (0, 2)]
pohyb(souradnice, 'j')
print(souradnice)  # → [(0, 0), (0, 1), (0, 2), (1, 2)]
pohyb(souradnice, 's')
print(souradnice)  # → [(0, 0), (0, 1), (0, 2), (1, 2), (1, 1)]
```

Funkce by neměla nic vracet. Nezapomeň na testy.

15. Napiš cyklus, který se bude ptát uživatele na světovou stranu, a podle ní zavolá pohyb, vykreslí seznam jako mapu, opět se zeptá na stranu, ...

```
Začínej se seznamem [(0, 0), (1, 0), (2, 0)]
```

- 16. Doplň funkci pohyb tak, aby při pohybu umazala první bod ze seznamu souřadnic. Výsledný seznam tak bude mít stejnou délku, jako před voláním. Uprav testy.
- 17. Doplň funkci pohyb tak, aby zamezila
 - pohybu ven z mapy
 - pohybu na políčko, které už v seznamu je

Vhodná výjimka pro tyto situace je ValueError('Game over'). Doplň i testy.

18. Přidej do hry hadí potravu. Tady jsou pravidla pro vegetariánského hada, ale můžeš si je změnit podle chuti:

Seznam ovoce obsahuje na začátku jedno ovoce, na políčku na kterém není had. (například: [(2, 3)] znamená jedno ovoce na pozici (2, 3).) Když had sežere ovoce, vyroste ("nesmaže" se mu ocas, tedy neprovede se to, cos přidala v úkolu 16), a pokud na mapě zrovna není další ovoce, na náhodném místě (kde není had) vyroste ovoce nové.

Každých 30 tahů vyroste nové ovoce samo od sebe.

Na mapě se toto tajemné ovoce zobrazuje jako otazník (?).

19. Hadí hřiště může mít libovolné rozměry větší než 4×1. Třeba 20×20 nebo 10×30.

A nakonec úkol na přemýšlení.