

## Domácí úloha 10 (29.11. až 12.12.2021 – 4 body) – zpd1u10.py

- Napište program pro generování playlistu nejlepších skladeb (stop).
  - Každá stopa o sobě uchovává informace o svém názvu, délce a uživatelském hodnocení.
  - Kompletní seznam stop je v playlistu, ze kterého je možné nechat náhodně vygenerovat nový playlist, na kterém jsou stopy, které mají lepší hodnocení než zadaná hodnota a jejich celková délka bude přibližně odpovídat zadané délce.
  - Pokud není dostatek stop s odpovídajícím hodnocením, program o tom uživatele informuje.
- Aplikace se bude skládat z následujících tříd a veřejných metod (případně další třídy či soukromé metody vytvářejte dle potřeby).
- Pomocí dokumentačních komentářů úlohu řádně okomentujte.
  - Komentář by měl být u každé třídy, metody a atributu (třídy i instance).
  - Pro třídy uveďte i autora a verzi (jako verzi uveďte aktuální datum ve tvaru RRRMMDD).
- Třída `Stopa`
  - Třída uchovává informace o jedné stopě, tedy název, délku (v sekundách), hodnocení (1.0 – nejhorší až 5.0 – nejlepší).
  - Veřejné metody
    - `def __init__(self, nazev, delka, hodnoceni):`
      - Super metoda vytvoří instanci stopy.
    - `def getDelka():`
      - Metoda vrátí délku stopy v sekundách.
    - `def getHodnoceni():`
      - Metoda vrátí hodnocení stopy.
    - `def toString():`
      - Metoda vrátí informace o stopě ve formátu:
        - název [čas] (hodnocení - počet hvězdiček/teček)
          - Čas je délka stopy ve formátu `Minuty: Sekundy`, a to i pro stopy s délkou nad hodinu (např. `62: 25`).
          - Počet hvězdiček je dán celou částí hodnocení. Pro desetinnou část hodnocení se vygeneruje znak `'.'`,
          - pokud je větší nebo rovna `0,25` a znak `'*'`, pokud je větší nebo rovna `0,75`. Např.:
            - hodnocení `1.20`: `*`
            - hodnocení `2.25`: `**`.
            - hodnocení `3.70`: `***`.
            - hodnocení `4.75`: `*****`
- Třída `Playlist`
  - Obsahuje seznam stop a informaci o jejich celkové délce.
  - Umožňuje vygenerování výběru podle zadaných kritérií.
  - Veřejné metody

- `def __init__(stopy):`
    - Super metoda vytvoří z pole stop playlist.
    - Může nastat i situace, že pole bude mít délku 0.
  - `def getCelkovaDelka():`
    - Metoda vrátí celkovou délku všech stop v playlistu.
  - `def filtrujHodnoceni(minHodnoceni):`
    - Metoda vygeneruje z originálního playlistu nový, který obsahuje pouze stopy s hodnocením větším či rovno zadanému parametru.
    - Originální playlist musí zůstat nezměněný.
  - `def vyberCelkovouDelku(minDelka):`
    - Metoda vygeneruje z originálního playlistu nový, který obsahuje přesně tolik stop (od začátku playlistu), aby jejich celková délka byla větší nebo rovna zadanému parametru a neobsahovala žádnou stopu navíc.
    - V případě že stop není dostatek, vrátí metoda všechny stopy.
    - Např. z playlistu obsahujícího 3 stopy, každá s délkou 10 sekund, budou pro
      - `minDelka == 15` vybrány první dvě stopy,
      - `minDelka == 40` vybrány všechny 3 stopy.
    - Originální playlist musí zůstat nezměněný.
  - `def zamichej():`
    - Metoda vygeneruje z originálního playlistu nový tým, že náhodně přeháže pořadí stop (pro každé volání musí být jiný výsledek).
    - Tip: náhodného pořadí lze docílit například tak, že pro každý index vygenerujete druhý náhodný index, a položky vzájemně prohodíte.
    - Originální playlist musí zůstat nezměněný.
  - `def toString():`
    - Vrátí textovou reprezentaci playlistu, kde na samostatných řádkách budou uvedeny jednotlivé stopy a poslední řádka bude obsahovat celkový čas playlistu ve formátu [Minuty:Sekundy] (i pro playlisty překračující jednu hodinu uvádějte pouze minuty a sekundy např. [87:25])
- Třída `Ppa1u10`
    - Spouštěcí třída.
    - Veřejné metody
      - `def main(minHodnoceni, minDelka):`
        - Vstupní metoda programu.
        - Vytvoří playlist zavoláním metody `vytvorPlaylist()`.
        - Z parametrů zjistí minimální požadované hodnocení a minimální požadovanou celkovou délku stop.
        - S pomocí metody `vyberStopy()` vygeneruje odpovídající playlist a vytiskne jej na obrazovku.
        - V případě, že délka vybraných stop není dostatečná, program o tom informuje hláškou: POZOR! Playlist je kratky
      - `def Playlist vytvorPlaylist():`
        - Metoda načte z počet stop a následně pro každou stopu její název, délku a hodnocení a z těchto stop vytvoří playlist.
      - `def vyberStopy(minHodnoceni, minDelka, playlist):`
        - Metoda na základě parametrů `minHodnoceni` a `minDelka` vybere z dodaného playlistu odpovídající stopy.

- Nejprve vybere stopy, které splňují minimální hodnocení (větší či rovno).
- Následně náhodně změní jejich pořadí.
- Nakonec vybere ze začátku playlistu minimální počet stop, s požadovanou minimální celkovou délkou (větší či rovno).

## Příklady

- Pro ušetření času při testování je vhodné použít přesměrování standardního vstupu ze souboru.
- Vstup programu
  - Zadán ručně po spuštění nebo zapsán v souboru při použití přesměrování vstupu
  - Použit pro oba následující příklady

4

With You

203

3.25

Points of Authority

200

3.5

By Myself

190

3.75

In the End

216

4.15

- **Příklad 1**
  - Vstup (načtení parametrů od uživatele)
    - hodnoceni = 3.5
    - delka = 391 (tj., 6:31)
  - Výstup
    - Protože je řazení náhodné, může výstup vypadat např. následovně:

In the End [3:36] (\*\*\*\*)

Points of Authority [3:20] (\*\*\*.)

[6:56]

- Do playlistu se nedostala stopa „By Myself“, protože délka prvních dvou vybraných stop je dostatečná.
- Při druhém spuštění bude seznam vypadat jinak, např.:

By Myself [3:10] (\*\*\*\*)

Points of Authority [3:20] (\*\*\*)

In the End [3:36] (\*\*\*\*)

[10:06]

- **Příklad 2**

- Vstup (načtení parametrů od uživatele)
  - hodnoceni = 3.75
  - delka = 410
- Výstup

By Myself [3:10] (\*\*\*\*)

In the End [3:36] (\*\*\*\*)

[6:46]

POZOR! Playlist je kratky