

A feladat megoldására és benyújtására 120 perc áll rendelkezésre. A létrehozott projekt megnevezése tartalmazza az Ön nevét, Neptun kódját és a feladatsor csoportját (pl. AliceBob_ABC123_X). A megoldását tartalmazó mappát (a teljes solution-t) tömörítve, a <http://zh.nik.lan> címen elérhető felületen keresztül nyújtsa be.

Ügyeljen a fordítási hibától mentes kódra, ellenkező esetben a megoldás sajnos nem értékelhető.

Készítsen szoftvert a „Gravity Falls” című sorozat epizódjainak elemzésére! A különböző *Console* kiírások és adatbkeresések helye a *Program* osztály. A további osztályokban/metódusokban **NE** használja a *Console* osztály metódusait (Pl. *ReadLine*, *WriteLine*).

1 Elkészítettünk az ön számára egy *Episode* osztályt. Az osztály rendelkezik néhány jellemzővel (*Sorszám*, *évad*, *éva*don belüli rész, *cím*, *értékelés*, *premier*, *rendező*). Továbbá elkészítettünk az ön számára egy *SeriesDatabase* osztályt is. Ez az osztály 1 darab belső *Episode* tömböt tartalmaz, ide kerülnek a *gravity.csv* fájlból az epizódok. A *SeriesDatabase*-ből egy példány van a *Program* osztályban. Nincs szükség további adatbázisok példányosítására.

A *gravity.csv* fájl egyes sorai (az elsőt kivéve, amely a fejléc) az epizódok jellemzőit tartalmazzák ; -el elválasztva. A *SeriesDatabase* osztály konstruktorában látja meghívva az alábbi két metódust:

- *ReadFile(path)*: Ennek a törzsében tölts fel a fájl sorai alapján az *Episode* tömböt.
- *DummyData*: Amennyiben nem tudja feltölteni, akkor ezt a metódust is használhatja.

Csak az egyiket hívja meg a konstruktorból! Értelmszerűen a *DummyData* használata 0 pontot ér, de legalább tovább tud haladni!

(7 pont)

2 *EpisodeType* felsorolás

- Hozza létre az *EpisodeType* felsorolás típust, amely egy rész lehetséges típusait tartalmazza! (*Normal*, *SeasonOpening*, *SeasonFinale*)

(2 pont)

3 *Episode* osztály

- Készítsen az osztálynak egy olyan publikus tulajdonságot amely írható és olvasható, típusa pedig *EpisodeType*!

(2 pont)

4 *SeriesDatabase* osztály

- Készítsen az osztályban egy visszatérési érték nélküli privát *Initialize* metódust. A metódusban járja be a belső tömböt és minden epizódnak állítsa be az előbb elkészített *EpisodeType* tulajdonságát a megfelelő értékre. Ezt a metódust hívja meg az osztály konstruktorának végén!

(9 pont)

- Készítsen egy olyan publikus double *AverageRating* tulajdonságot, amely csak olvasható és megadja az összes rész átlagos értékelését két tizedesjegy pontosan!

(5 pont)

- Készítsen egy publikus *WorstEpisode* metódust, amely visszaadja a legrosszabb értékelésű epizódot!

(4 pont)

- Készítsen egy publikus *SeasonData* metódust, amely visszaadja tömbként az évad legfontosabb információit az alábbi formátumban:

```
Évad adatok:  
1. évad : 20 rész. Vetítve: 265 napig.  
2. évad : 21 rész. Vetítve: 563 napig.
```

(8 pont)

- Az előző metódus által visszaadott tömböt lehessen kiírni a megadott fájlba a publikus *Save(fileName, data)* metódus segítségével! Amennyiben létezik már ilyen nevű fájl, ne történjen meg a mentés, ezt pedig a visszatérési érték segítségével jelezzük!

(5 pont)

- Készítse el a *Director* osztályt a következő, kívülről csak olvasható jellemzőkkel:

- Name (szöveg)
- Episodes (*Episode* objektumokból álló gyűjtemény)

Az osztály szabadon bővíthető konstruktorral, metódusokkal, amennyiben szükséges!

(2 pont)

- Készítsen egy publikus *DirectorDetails* metódust, amely visszatér a rendezők tömbjével. Egy rendező csak egyszer szerepeljen a tömbben. A rendező által rendezett részek ismétlődés és null elemektől mentesen szerepeljenek a Rendező belső gyűjteményében.

(16 pont)

- Hozza létre a publikus *GravityDays* metódust, amely visszatér a legelső és legutolsó rész között eltelt napok számával!

(4 pont)

5 Program osztály

- Tesztelje az összes elkészített publikus tulajdonságot és metódust a *Main*-ben!

(6 pont)