

Dokumentáció – Filmtár

A feladat megoldásakor a filmeket tároló osztály és azon belül a fájlból olvasás/fájlba írás függvények voltak nagyobb kihívások.

Alapvetően van egy Film osztály, amit örököl a Csaladi és a Dokumentum osztály. Ezek kiegészítik azt több tárolható adattal, és egyes függvényeken minimálisan változtatnak.

A Filmtar nevű osztály meg képes ezek eltárolására, a benne lévő *Film*** típusú pointer használatával.

Class: Filmtar

Paramérerei:

int db: Filmtár mérete

Film** filmek: Film típusú objektumokra mutató pointerekre mutató pointer. Így tároljuk egy könnyen elérhető helyen a Film és az abból származtatott osztályok (Csaladi, Dokumentum, stb.) objektumait)

Fontosabb függvényei:

Filmtar()

Paraméter nélküli konstruktor, mindent alapértéken hagy.

void beolvas(const char fajlnev)*

Beolvassa a filmtárat egy adott fájlból. Void fv., tehát nem ad vissza semmit, de paraméterként megadható a fájl neve.

Filmtar(const char fajlnev)*

Konstruktor amely azonnal be is olvassa a fájlt. Paramétere szintén a fájl neve.

Void elment(const char fajlnev)*

A megadott nevű fájlba elmenti a Filmtárat, ugyanolyan formátumban mint ahogy beolvasta.

Void any_print(std::ostream& os, Film f)*

Meghívja a megfelelő osztály print() függvényét.

Void list(std::ostream& os)

Adott output streamre kilistázza az összes filmet.

*Filmtar keres(char const *s)*

Megadott szöveggel lehet keresni a Filmtarban, szintén Filmtarat ad vissza.

Void hozzáad(Film f)*

Film hozzáadása a filmtárhoz. Az objektumra mutató pointert kapja paraméterként.

Void torol(Film f)*

Film törlése a filmtárból. A hozzáadással azonos elven működik.

Void rendez()

Ábécé sorrendbe rendezi a filmeket a címük alapján. Egy szimpla bubble sort.

Class: Film

Paramérerei:

string cim;

A film címe.

int hossz;

A film hossza másodpercben.

int ev;

A film kiadásának éve.

Fontosabb függvényei:

3 fajta konstruktor: paraméter nélküli, másoló, paraméteres

Virtual int tipus() const

Visszaadja a film típusát. Virtuális mert az alosztályoknál más a típus.

Int getHossz()

Int GetEv()

String getCim()

Getterek az osztály adataihoz.

virtual void print(ostream& os)

Kiírja a film adatait egy adott output streamre.

Class: Csaladi

Paramétereit:

Int korhatar;

A film korhatára, egész szám.

Fontosabb függvényei:

3 fajta konstruktor: paraméter nélküli, másoló, paraméteres

Ezek közül mind meghívja az eredeti *Film* osztály konstruktorát is.

Int getKorhatar() const

Visszaadja a film korhatárát.

Void print(ostream& os) const

Kiírja a film adatait egy adott output streamre. Annyi az eltérés az alaposztálytól, hogy kiírja a korhatárt is.

Class: Dokumentum

Paramétereit:

String leiras;

Szöveges leírás a dokumentumfilmhez.

Fontosabb függvényei:

3 fajta konstruktor: paraméter nélküli, másoló, paraméteres

Ezek közül mind meghívja az eredeti *Film* osztály konstruktorát is.

String getLeiras() const

Visszaadja a film leírását.

Void print(ostream& os) const

Kiírja a film adatait egy adott output streamre. Annyi az eltérés az alaposztálytól, hogy kiírja a film leírását is.

A konzolba elvárt kimenet a tesztkörnyezet futtatásakor (a *Filmtar* osztály tesztéseinél):

```
Keresztapa      2000      1980
Matrix  20      20
Avatar   3123    18
Star Wars I    20      20
Star Wars II   20      20
Star Wars III  20      20
Star Wars IV   20      20
Star Wars V    20      20
Star Wars VI   20      20
Teszt    20      20
ZZzzzz   50      20      16
Aaaaaa   201     3242    Leiras a dokumentumfilmhez
Rendezes utan:
Aaaaaa   201     3242    Leiras a dokumentumfilmhez
Avatar   3123    18
Keresztapa      2000      1980
Matrix  20      20
Star Wars I    20      20
Star Wars II   20      20
Star Wars III  20      20
Star Wars IV   20      20
Star Wars V    20      20
Star Wars VI   20      20
Teszt    20      20
ZZzzzz   50      20      16

Process returned 0 (0x0)   execution time : 0.038 s
```

Filmtár – eredeti specifikáció

1. Feladat

Készítsen filmeket nyilvántartó rendszert. Minden filmnek tároljuk a címét, lejátszási idejét és kiadási évét. A családi filmek esetében korhatár is van, a dokumentumfilmek esetében egy szöveges leírást is tárolunk. Tervezzen könnyen bővíthető objektummodellt a feladathoz! Demonstrálja a működést külön modulként fordított tesztprogrammal! A megoldáshoz **ne** használjon STL tárolót!

2. Feladatspecifikáció

Egy program melynek alapvető célja filmek adatainak nyilvántartása egy adatbázisban. A filmeknek eltároljuk a címét, lejátszási idejét, kiadási évét.

A *családi filmek* esetében korhatár is van, a *dokumentumfilmek* esetében egy szöveges leírást is tárolunk. Fontos, hogy a program új film műfajokkal könnyen bővíthető legyen. Ehez használunk öröklést az osztályok között. A memóriában az adatmezőket más-más adattípusként tároljuk. Például a film címe string, de a korhatár egész szám. A program csak angol karakterek beolvasására és tárolására alkalmas.

Az adatbázist egy *.txt fájlként tároljuk és egy külön függvénnyel be kell hogy tudjuk olvasni memóriába, egy másikkal pedig elmenteni. Ha nem létezik a fájl, akkor a beolvasás üres adatbázist ad vissza, ami pedig ugyanúgy módosítható lesz.

A fájlban az adatokat (filmeket) külön sorokba írjuk, míg az adatmezőket tabulátorok ('\\t' karakterek) választják el. Mindez biztosítja a filmtár adatbázisának kompatibilitását más adatbázis- vagy táblázatkezelő szoftverrel.

Lehet függvényekkel új film adatait hozzáadni, vagy egy meglévő filmét törölni.

Szintén le lehet szűrni a filmeket egy keresett karaktersorozat megadásával. Ez a művelet visszaadja az összes találatot.

Az adatbázis rendezésére és egy film vagy az egész adatbázis konzolban való kiírására szintén külön függvényeket készítünk.

Külön tesztelőmodult kell létrehozni, amely bemutatja a program működését. Létrehoz új adatokat és azokkal teszteli az összes elvégezhető műveletet, figyelemmel speciális esetekre amik hibát okozhatnak.