

Szoftverfejlesztő – OKJ 54 213 05	
Komplex alkalmazás készítése	Időtartam: 180 perc

GYAKORLATI VIZSGATEVÉKENYSÉG – PRÓBAVIZSGA

ÁLTALÁNOS TUDNIVALÓK

- Készítsen grafikus alkalmazást, amellyel elvégezhető az alábbi feladatok!
- Az alkalmazás elkészítéséhez tetszőleges (nem webes) fejlesztői környezetet, illetve programozási nyelvet használhat!
- Segítségül kézzel írott saját jegyzeteket, kézikönyveket, illetve a gépre telepített offline help rendszereket használhat!
- **Kiadott források:**
 - konferencia.txt – az input állomány
 - Kepek.zip – a feladat megoldásához szükséges képek
- **Beadandó:** a teljes projekt, illetve a megoldás során létrehozott output állomány.
 - Mindezeket egyetlen tömörített fájlban küldje el a vizsgáztató által megadott e-mail címre!
 - Az állomány neve gyakvizsgakomplex2018_sajatnev_osztaly legyen!
 - A tárgyba is ugyanezt írja!
 - A beadott alkalmazás futtatható állapotú legyen!
 - A hibás vagy hiányos részeket kommentben hagyja a kódban, de jelezze, hogy az is a megoldás része!
 - A gyorsabb értékeléshez az elkészített programot futtatható állapotban hagyja megnyitva a számítógépén!

FELADATLEÍRÁS

Egy irodalomtanárok számára tartott konferencián egyidejűleg több előadást is tartottak, különböző termekben.

- A vendégek a meghallgatott előadásokat 1 és 3 közötti pontszámmal értékelhették, amelyet a helyükön elhelyezett szavazókészülékkel tudtak elküldeni.

Az egyes előadások értékelését a *konferencia.txt* állomány tartalmazza, ebből egy részlet látható itt:

```
Babits Mihály gondolati lírája (Tóth Miklós)
3;6
2;3;2;1;1;1
3;2;2;1;0;1
1;1;2;1;2;0

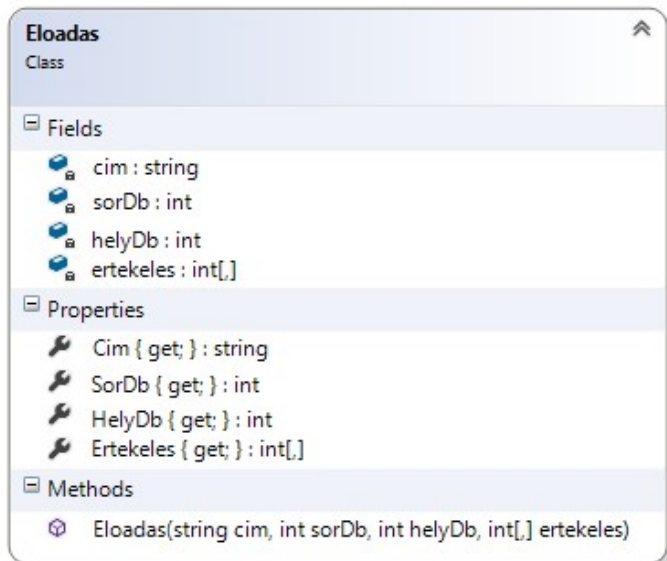
Petőfi Sándor hatása az utókorra (Kovács Pál)
5;6
2;1;1;3;3;3
3;3;1;1;2;2
1;1;1;2;2;1
3;3;3;1;1;2
3;3;3;3;3;1
```

A fájl több szakaszból áll, amelyet üres sorok választanak el.

- A szakaszok első sora az előadás címe, utána zárójelben az előadó neve
- A következő sorban az előadóterem sorainak és a sorokban lévő helyek száma található, pontosvesszővel elválasztva. Pl. a Babitsról szóló előadás termében 3 sor van, minden sorban 6 hely.
- Az utána következő sorokban pontosvesszővel elválasztva egész számok találhatók, melyek értéke 0, 1, 2, vagy 3 lehet.
- A számértékek jelentése:
 - 0 – az adott helyen nem ült senki
 - 1 – az adott helyen ülő vendég 1-esre értékelte az előadást
 - 2 – az adott helyen ülő vendég 2-esre értékelte az előadást
 - 3 – az adott helyen ülő vendég 3-asra értékelte az előadást
- A könnyebb feldolgozás céljából a fájl utolsó sora egy ÜRES SOR!

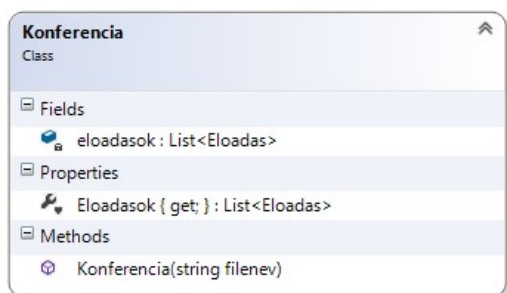
1. feladat – Az adatszerkezet kialakítása, a szövegfájl beolvasása

- a) Hozzon létre egy új projektet **KonferenciaProjekt** néven!
- b) Hozzon létre egy **Eloadas** osztályt egy előadás adatainak a kezeléséhez!



- A **cim** adattag az előadás címét tárolja.
 - A **sorDb** adattag az előadóterem sorainak a számát tárolja.
 - A **helyDb** adattag a sorokban található helyek számát tárolja.
 - Az **ertekeles** adattag egy kétdimenziós tömb, egész számokat tartalmaz, melyek értéke 0, 1, 2 vagy 3 lehet. A tömb *i.* sorának *j.* eleme az *i.* sor *j.* nézőjének az értékelése.
- Az adattagokhoz készítsen get property-t, vagy getter metódusokat!
- Hozzon létre az **Eloadas** osztályban paraméteres konstruktort, amely a paraméterek értékével inicializálja az adattagokat!

- c) Hozzon létre egy **Konferencia** osztályt az összes előadás adatainak a kezeléséhez!



- Az **eloadasok** adattag jelentése:
- Az **eloadasok** adattag egy **Eloadas** típusú objektumokat tartalmazó lista, amely lehetővé teszi a konferencia összes előadásának a kezelését.
- Készítsen az adattaghoz get propertyt, vagy getter metódust!
- Írjon a **Konferencia** osztályban konstruktort, amely beolvassa a paraméterében megadott szövegfájlt, és annak tartalma alapján feltölti az **eloadasok** listát az értékelésekkel.
- A program indításakor a **Konferencia** osztály kerüljön példányosításra a **konferencia.txt** fájl tartalma alapján.
- A **Konferencia** típusú objektumpéldányt olyan módon hozza létre, hogy az a grafikus megjelenítést végző osztályban elérhető legyen!
- Ha a fájl beolvasása során bármilyen hiba adódik, a program adjon hibaüzenetet, és a program futása szakadjon meg!

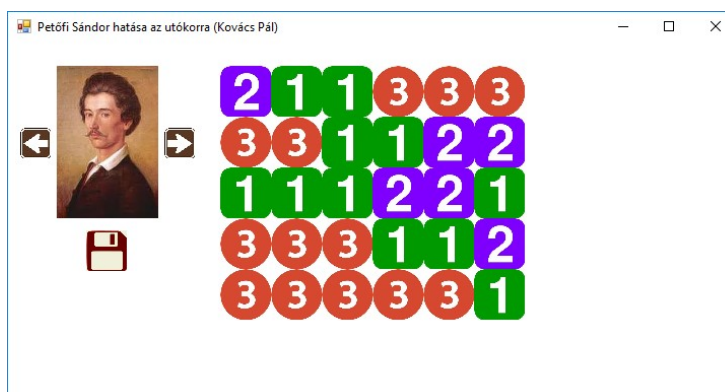
Szoftverfejlesztő – OKJ 54 213 05	
Komplex alkalmazás készítése	Időtartam: 180 perc

2. feladat – Az előadások értékelésének a megjelenítése

- a) A projekt grafikus megjelenítést végző osztályában hozza létre a felület elemeit úgy, hogy az alábbi mintához hasonló megjelenítést tegyen lehetővé!



- Az ablak bal oldalára helyezzen a mintának megfelelő elrendezésben olyan komponenseket, amelyeken megjeleníthető
 - annak az írónak a képe, akiről az előadás szól
 - a navigációs nyilak, (*bal.jpg*, *jobb.jpg*)
 - illetve a mentés ikon (*ment.jpg*)
 - Helyezzen ezek mellé egy tároló komponenst (panelt), amelyre az előadás értékelését jelző képeket tesz majd a program.
- b) Írjon egy metódust ugyanebben az osztályban, amely a paraméterében megadott sorszámú előadás értékeléseit megjeleníti, a beolvasott adatok alapján, a fenti mintának megfelelően!
- Jelenítse meg annak az írónak a képét, akiről az előadás szól
 - A mellékelt fájlok között megtalálható az összes író képe, a fájlok neve megegyezik az írók vezetéknévvel (pl. *Petőfi.jpg*, *Móricz.jpg*)
 - Jelenítse meg az előadás címét, és az előadót az ablak fejlécében!
 - Az értékelések megjelenítéséhez használja a mellékelt képek közül az alábbiakat!
 - *Pont0.jpg* (szürke) – az adott helyen nem ült senki
 - *Pont1.jpg* (zöld) – az adott helyen ülő 1-esre értékelte az előadást
 - *Pont2.jpg* (kék) – az adott helyen ülő 2-esre értékelte az előadást
 - *Pont3.jpg* (piros) – az adott helyen ülő 3-asra értékelte az előadást
 - A program indítása után az első előadás értékelése legyen látható! (Ld. a fenti mintát)
- c) Tegye lehetővé az előadások közötti lapozást!
- A jobb nyílra kattintva jelenjen meg a következő, a balra nyílra kattintva pedig az előző előadás értékelése!
 - A lapozás körkörös legyen, azaz
 - az utolsó előadásról jobbra lapozva az első előadásra léphessünk,
 - az első előadásról balra lapozva az utolsó előadásra léphessünk!
 - Pl. a Babitsról szóló előadásról jobbra lapozva a Petőfi előadásra kell lépnünk!



Szoftverfejlesztő – OKJ 54 213 05	
Komplex alkalmazás készítése	Időtartam: 180 perc

3. feladat – Módosítás és mentés

- a) Tegye lehetővé az egyes értékelések módosítását!
- Ha egy helyre bal egérgombbal rákattintunk, akkor az adott értékelés a következők szerint változzon:
 - 1 \Rightarrow 2
 - 2 \Rightarrow 3
 - 3 \Rightarrow 1
 - Bármelyik szürkén látható helyre kattintva ne történjen változás, és a program jelenítsen meg üzenetet, hogy az adott helyen nem ült senki!
 - A lapozás során a változások maradjanak meg!
- b) Tegye lehetővé a módosítások mentését!
- A Mentés ikonra kattintva a módosítások kerüljenek elmentésre a `konferencia.txt` fájlban!
 - A mentés előtt készítsen biztonsági másolatot a `konferencia.txt` fájlról, `konferencia.bak` néven!
 - A sikeres/sikertelen mentésről a program adjon visszajelzést!

4. feladat – Előadás átlagos értékelésének a megjelenítése

- a) Bővítse az **Eloadas** osztályt egy metódussal, amellyel meghatározható az egyes előadások átlagos értékelése!
- b) Lapozáskor jelenítse meg az előadás átlagos értékelését az alábbi mintának megfelelően!
- Az átlagba csak a tényleges értékeléseket számolja bele!
- c) Módosításkor az átlag értéke is azonnal változzon meg a fejlécben!

