

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:**
  - petrikgepek.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 gyakvizsgakomplex2021\_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

A Petrik Lajos technikum géptermeit felújították. A géptermekek nevét is kaptak, és minden számítógépre felkerült egy új image. Az imagek tesztelése egyidejűleg történt, több osztály bevonásával.

- A diákok a feltelepített imageket 1 és 3 közötti pontszámmal értékelhették, amelyet a helyükön elhelyezett szavazókészülékkel tudtak elküldeni.

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

```
Neumann János gépterem (5/13T)
3;6
2;3;2;1;1;1
3;2;2;1;0;1
1;1;2;1;2;0

Gordon Moore gépterem (2/14R)
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 a gépterem neve, utána zárójelben az értékelő osztály
- A következő sorban a gépterem sorainak és a sorokban lévő helyek száma található, pontosvesszővel elválasztva. Pl. a Neumannról elnevezett teremben 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 diák 1-esre értékelte az image
  - 2 – az adott helyen ülő diák 2-esre értékelte az image
  - 3 – az adott helyen ülő diák 3-asra értékelte az image
- A könnyebb feldolgozás céljából a fájl utolsó sora egy ÜRES SOR!

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

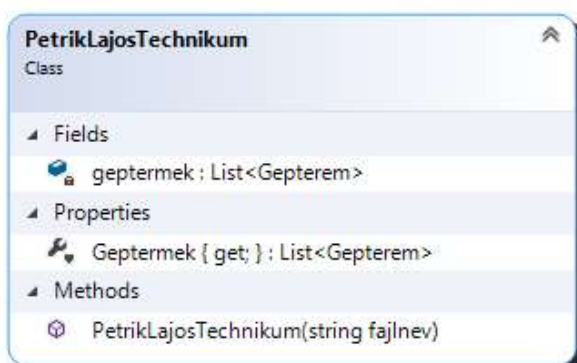
### 1. feladat – Az adatszerkezet kialakítása, a szövegfájl beolvasása

- Hozzon létre egy új projektet **GepteremProjekt** néven!
- Hozzon létre egy **Gepterem** osztályt egy gépterem adatainak a kezeléséhez!



- A **nev** adattag a gépterem nevét tárolja.
  - A **sorDb** adattag a gépterem 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. diákjának az értékelése.
- Az adattagokhoz készítsen get property-eket, vagy getter metódusokat!
- Hozzon létre a **Gepterem** osztályban paraméteres konstruktort, amely a paraméterek értékével inicializálja az adattagokat!

- Hozzon létre egy **PetrikLajosTechnikum** osztályt az összes gépterem adatainak a kezeléséhez!



- A **gepteremek** adattag jelentése:
- A **gepteremek** adattag egy **Gepterem** típusú objektumokat tartalmazó lista, amely lehetővé teszi az iskola összes gépteremének a kezelését.
- Készítsen az adattaghoz get propertyt, vagy getter metódust!
- Írjon a **PetrikLajosTechnikum** osztályban konstruktort, amely beolvassa a paraméterében megadott szövegfájlt, és annak tartalma alapján feltölti a **gepteremek** listát az értékelésekkel.
- A program indításakor a **PetrikLajosTechnikum** osztály kerüljön példányosításra a **petrikgepek.txt** fájl tartalma alapján.
- A **PetrikLajosTechnikum** 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!

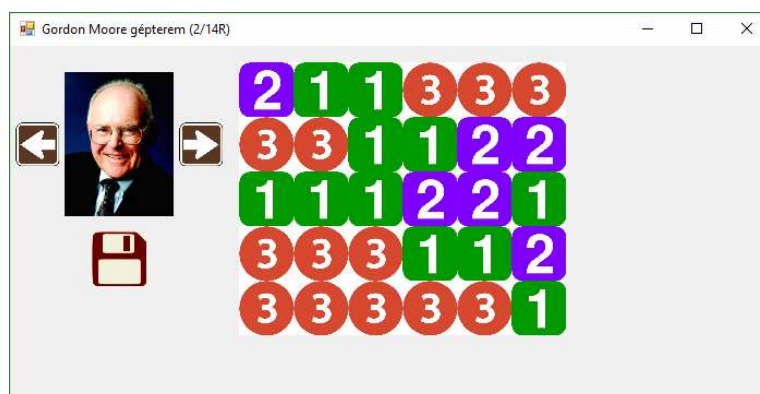
Szoftverfejlesztő – OKJ 54 213 05	
Komplex alkalmazás készítése	Időtartam: 180 perc

## 2. feladat – A gépterem 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 a személynek a képe, akiről a gépterem a nevét kapta
    - a navigációs nyilak, (*bal.jpg*, *jobb.jpg*)
    - illetve a mentés ikon (*ment.jpg*)
  - Helyezzen ezek mellé egy tároló komponenst (panelt), amelyre a gépterem értékelését jelző képeket tesz majd a program.
- b) Írjon egy metódust ugyanebben az osztályban, amely a paraméterében megadott sorszámú gépterem értékeléseit megjeleníti, a beolvasott adatok alapján, a fenti mintának megfelelően!
- Jelenítse meg annak a személynek a képét, akiről a gépterem a nevét kapta
    - A mellékelt fájlok között megtalálható az összes személy képe, a fájlok neve megegyezik a személyek első nevével (pl. *Neumann.jpg*, *Bill.jpg*)
  - Jelenítse meg a gépterem nevét, és az értékelő osztályt 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 imaget
    - *Pont2.jpg* (kék) – az adott helyen ülő 2-esre értékelte az imaget
    - *Pont3.jpg* (piros) – az adott helyen ülő 3-asra értékelte az imaget
  - A program indítása után az első gépterem értékelése legyen látható! (Ld. a fenti mintát)
- c) Tegye lehetővé a gépterem közötti lapozást!
- A jobb nyílra kattintva jelenjen meg a következő, a bal nyílra kattintva pedig az előző gépterem értékelése!
  - A lapozás körkörös legyen, azaz
    - az utolsó gépteremről jobbra lapozva az első gépteremre léphessünk,
    - az első gépteremről balra lapozva az utolsó gépteremre léphessünk!
  - Pl. a Neumannról elnevezett gépteremről jobbra lapozva a Moore-ról elnevezettre kell lépünk!



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	⇒	2
• 2	⇒	3
• 3	⇒	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 `petrikgepek.txt` fájlban!
  - A mentés előtt készítsen biztonsági másolatot a `petrikgepek.txt` fájlról, `petrikgepek.bak` néven!
  - A sikeres/sikertelen mentésről a program adjon visszajelzést!

### 4. feladat – Image átlagos értékelése

- a) Bővítse a **PetrikLajosTechnikum** osztályt egy metódussal, amellyel meghatározható az image átlagos értékelése.
- Az átlagba csak a tényleges értékeléseket számolja bele, az üres helyeket ne!
  - Az átlagba az összes gépteremből érkezett értékelést számolja bele.
- b) Egészítse ki a felületet egy nyomógommbal, amivel az átlagos értékelést tudja megjeleníteni az alábbi mintának megfelelően!
- c) Gombkattintásra jelenjen meg egy felugró ablak, ami megjeleníti a teljes átlagos értékelést, illetve kiírja, hogy kell-e új imageket készíteni, a lenti mintának megfelelően.
- Az imagek csak akkor mennek át a tesztelésen, ha az átlagos értékelés eléri a 2.0-t. Ha az értékelés ez alatt van, akkor új imagek fognak készülni.

