

2. Programozás

40 pont

Iskola²

A `nevek.txt` állományban rögzítettük egy középiskola tanulóinak néhány adatát. Feltételezheti, hogy nincs két azonos nevű tanuló egy osztályban. Az állomány tartalma soronként:

- iskola kezdésének éve (2004-2007)
- az osztály betűjele (a-e)
- a diák neve (ékezetek nélkül).

Az adattagok pontosvesszővel vannak elválasztva. Példa (részlet) a `nevek.txt` állományra:

```
2004;d;Vavrek Kristof  
2006;e;Hidas Reka  
2006;d;Kun Michael
```

Megoldásban:

- *A képernyőre írást igénylő részfeladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például: 3. feladat)!*
- *Az egyes feladatokban a kiírásokat a minta szerint készítse el!*
- *Az ékezetmentes kiírások is elfogadottak.*
- *Az azonosítókat kis- és nagybetűkkel is kezdheti.*
- *A program megírásakor az állományban lévő adatok helyes szerkezetét nem kell ellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.*
- *A megoldását úgy készítse el, hogy az azonos szerkezetű, de tetszőleges bemeneti adatok mellett is helyes eredményt adjon!*

Oldja meg a következő feladatokat:

1. Készítsen `Iskola` néven **konzol típusú projektet**, melyben megoldja a következő feladatokat!
2. Olvassa be és tárolja el egy célszerűen megválasztott adatszerkezetben (pl.: vektor, lista) a `nevek.txt` állományban lévő adatokat!
3. Írja ki a képernyőre, hogy hány tanuló jár az iskolába!
4. Írja ki a képernyőre a minta szerint, hogy ki rendelkezik a leghosszabb névvel! A szóközöket ne számolja a név hosszához! Holtverseny esetén minden, leghosszabb névvel rendelkező tanuló neve jelenjen meg! Jelenítse meg a kiírásban a leghosszabb név/nevek hosszát is a minta szerint!
5. Az iskolai rendszergazdának egyedi azonosítókat kell készítenie a számítógép-hálózat használatához. Az azonosítókat a következő módon alakítja ki: első karaktere az évfolyam utolsó számjegye (pl.: 2006 esetén 6), következő karakter az osztály betűjele, majd a vezetéknév első három karaktere, végül első keresztnév első három karaktere következik. Az azonosítóban mindenütt kisbetűk szerepelnek. Feltételezhetjük, hogy a vezetéknév és az első keresztnév legalább 3 karakteres. Készítsen jellemzőt vagy függvényt, melyben meghatározza a rendelkezésre álló adatokból a tanuló azonosítóját! Az azonosítókat ne tárolja! Az elkészített jellemzőt/függvényt felhasználva írja ki az adatszerkezetben tárolt első és utolsó tanuló azonosítóját a minta szerint!

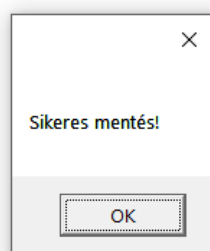
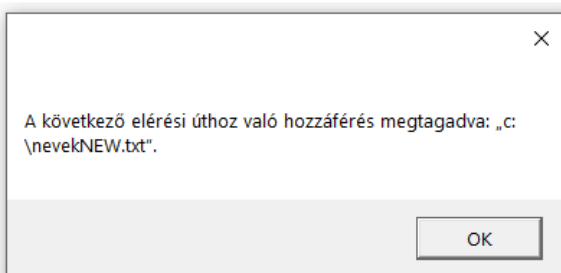
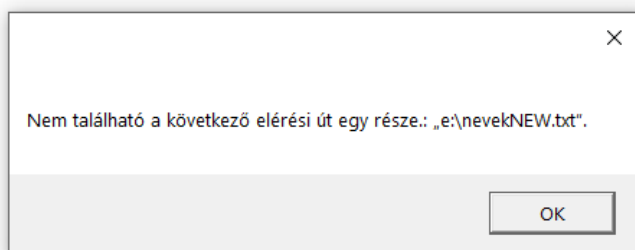
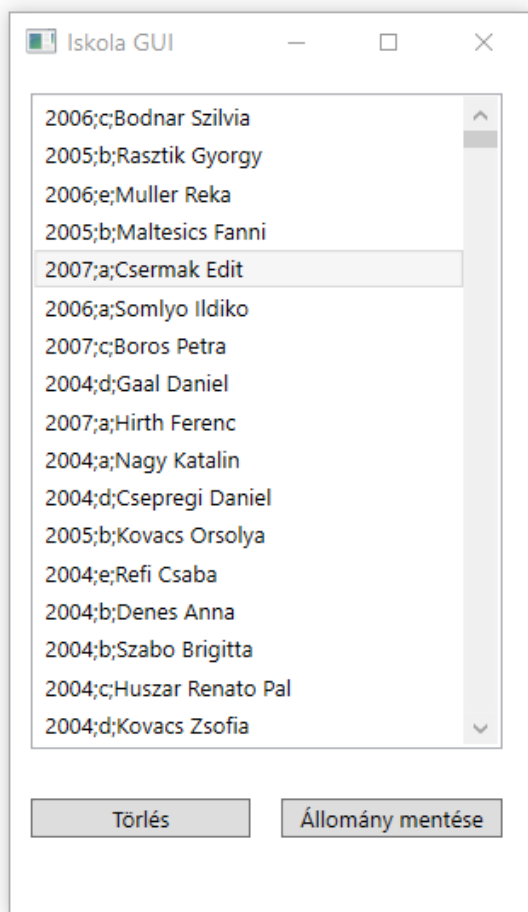
² Forrás: <https://docplayer.hu/4402962-Informatika-emelt-szintu-probaerettsegi.html>

6. Kérjen be egy azonosítót és állapítsa meg, hogy ki tartozhat hozzá! A tanuló adatait írja a képernyőre! Ha nem talál megfelelő diákot, akkor a „Nincs megfelelő tanuló.” mondatot jelenítse meg a minta szerint!
7. A forráskódjába a meglévő osztálydefiníció fölé illessze be a `java.txt` vagy a `csharp.txt` forrásállományból a `JelszóGeneráló` osztályt definiáló kódrészletet, majd hozzon létre belőle egy osztálypéldányt (objektumot)! Az osztálypéldány konstruktorát a véletlenszámok előállítására alkalmas beépített objektummal paraméterezze! Válaszon ki véletlenszerűen egy tanulót, majd a `JelszóGeneráló` osztály `Jelszó()` metódusának hívásával írjon ki egy 8 karakter hosszú jelszót a minta szerint!
8. Készítsen **grafikus alkalmazást** a következő feladatok megoldására, melynek projektjét `IskolaGUI` néven mentse el!
9. Az alkalmazás grafikus felhasználói felületét alakítsa ki a minta szerint! Az ablak címsorában a „Iskola GUI” szöveget jelenítse meg!
10. Az ablakon található listába a program induláskor töltse be a `nevekGUI.txt` állomány sorait! A lista elemei a forrásfájl egy-egy sora (továbbiakban tanulója) legyen!
11. Oldja meg, hogy a kijelölt tanuló a „Törlés” parancsgomb lenyomása után törlésre kerüljön a listából! Ha a listában nincs kijelölt tanuló, akkor törléskor a „Nem jelölt ki tanulót!” szöveg jelenjen meg egy felugró ablakban!
12. Ha az „Állomány mentése” parancsgombra kattintunk, akkor történjen meg a listából a tanulók mentése a `nevekNEW.txt` állományba, melynek szerkezete a forrásállomány szerinti legyen! Ha a mentés sikeres volt, akkor a „Sikeres mentés!” felirat jelenjen meg egy felugró ablakban! Ha az állomány mentése sikertelen, akkor a hibaüzenet (a hibához tartozó beépített üzenet/message) jelenjen meg egy felugró ablakban! Lehetséges hibaokokhoz tartozó beépített üzeneteket a minták között talál!

Minták a konzolos alkalmazás futására:

3. feladat: Az iskolába 650 tanuló jár.
4. feladat: A leghosszabb (25 karakter) nevű tanuló(k):
Karatson Patricia Mercedesz
5. feladat: Azonosítók
Első: Bodnar Szilvia - 6cbodsz
Utolsó: Krizsan Vivien Evelin - 6ckriviv
6. feladat: Kérek egy azonosítót [pl.: 4dvavkri]: 4atolrek
2004 a Tolcsvay-Nagy Reka Luca
7. feladat: Jelszó generálása
Szabo Mihaly - hn4xyutl
3. feladat: Az iskolába 650 tanuló jár.
4. feladat: A leghosszabb (25 karakter) nevű tanuló(k):
Karatson Patricia Mercedesz
5. feladat: Azonosítók
Első: Bodnar Szilvia - 6cbodsz
Utolsó: Krizsan Vivien Evelin - 6ckriviv
6. feladat: Kérek egy azonosítót [pl.: 4dvavkri]: 5cmogab
Nincs megfelelő tanuló.
7. feladat: Jelszó generálása
Tompos Zsolia - smr94dud

Minták a grafikus alkalmazás futására:



Java hibaüzenetek:

