

# 1. Programozás (Órarend)

Az Einsteinium Középiskola órarendjének adatait az iskola a termekben levő osztályokkal adja meg egy szöveges állományban. Ebben a feladatban ezen iskola órarendjének forrásfájljával kell dolgoznia, mellyel programozási feladatokat kell megoldania.

A feladat megoldás során vegye figyelembe a következőket:

- *A képernyőre írást igénylő részfeladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például: 5. feladat)!*
- *Az egyes feladatokban a kiírásokat a minta szerint készítse el!*
- *Az ékezetmentes azonosítók és kiírások is elfogadottak.*
- *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!*

A `termek.txt` pontosvesszővel tagolt forrásállomány sorai a következő módon épülnek fel:

- Az állományban az adatok jellege 7 soronként ismétlődik.
- Egy ilyen 7 adatsorból álló blokkban az első sor mindig a terem számának azonosítója (ez minden esetben három számjegyből áll). Az azonosító első számjegye jelöli, hogy a terem hányadik emeleten helyezkedik el.
- Az ezt követő **6 sorban** a sorok az órarend óráit jelölik, minden teremben az elsőtől a hatodikig óráig vannak megadva az adatok. Minden sorban **5 elem (oszlop)** található, az első a hétfői, a második a keddi, a harmadik a szerdai, a negyedik a csütörtöki, az ötödik sor pedig a pénteki nap óráit tartalmazza.
- A sorokban kétféle adat szerepelhet. Ha az adat 3 jegyű, akkor az első két karakter mindig az adott osztály évfolyamára utal (9-12. évfolyamig, a 9. évfolyam azonosítója 09), az utolsó nagybetű pedig A, B, C, D vagy E az osztályt azonosítja. Ha az adott időben a teremben nincsen óra, akkor ott egy 0-s számjegy szerepel. □ Például a 117-es teremben (ami az első emeleten van) hétfőn az első két órában a 10. B osztály tartózkodik, míg a 202-es teremben (ami a második emeleten található) pénteken az 1-től az 5-ödik óráig a 11. A osztálynak van órája.

```
117
10B;11B;10D;10A;09C
10B;10B;11D;10A;09C
10E;10B;11D;09A;11D
10E;10B;11D;09D;11D
09B;09B;11D;09D;09B
09B;12D;10B;10C;09C
202
11A;11B;10D;10A;11A
11A;10B;11D;10A;11A
10E;10B;11D;09A;11A
10E;10B;11D;09D;11A
09B;09B;11D;09D;11A
10A;12D;10B;10C;09C
```

1. Készítsen **konzol alkalmazást** a következő feladatok megoldására, amelynek projektjét orarend néven mentse el!
  2. A projekthez adjon hozzá új osztályt `Terem` néven!
  3. A `Terem` osztály konstruktora tárolja el a paraméter/paraméterekben átadott terem számát és a hozzá tartozó órarend mátrixot! Az adatokat a feladatok megoldásához célszerűen megválasztott típusú adattagokban (például: karakter, karakterlánc lista, karakterlánc vektor, karakter mátrix stb.) tárolja a `Terem` osztályban! A feladat megoldásához vizsgálja meg a `termek.txt` állomány felépítését! Feltételezheti, hogy az állományban nincs szerkezeti hiba.
  4. Olvassa be a `termek.txt` állományban található teremszámokat és órarendeket! A beolvasott adatokat tárolja a `Terem` osztály példányaiban egy célszerűen megválasztott adatszerkezetben (például: `Terem` osztály típusú listában).
  5. Határozza meg és írja ki a képernyőre, hogy hány terem adatai szerepelnek a forrásállományban!
  6. Kérjen be a felhasználótól egy évfolyamot a minta szerint! Az adatbevitelt ismételje mindaddig, amíg a bemenet nem a 9 és 12 közötti egész szám! Ügyeljen rá, hogy hibás adat esetén ne szakadjon meg a program futása!
  7. Határozza meg és írja ki a képernyőre, hogy az előző feladatban bekért évfolyamnak összesen hány órája van egy héten! Ha az előző feladatban nem sikerült a bekérés, akkor a 9-es értékkel dolgozzon! Ügyeljen arra, hogy a 9. évfolyamos osztályoknál a forrásfájlban a 09-es előtag használatos!
  8. Készítsen függvényt a főprogramban, amely paraméterben egy osztály azonosítóját kapja meg (a forrásállománynak megfelelő formátumban) és visszaadja, hogy hányadik emeleten van az osztálynak leggyakrabban órája! Tesztelésként hívja meg a függvényt a „12A” paraméterrel és írja ki, hogy az osztálynak hányadik emeleten van a legtöbb órája a minta szerint! Feltételezheti, hogy mindegyik osztály esetében egyértelműen meghatározható a feladat szerinti emelet.
  9. Az előző feladatban létrehozott függvény felhasználásával határozza meg és írja ki a képernyőre a mintának megfelelően, hogy hány olyan osztály van, akinek nem az 1. emeleten van a legtöbb órája! Feltételezheti, hogy legalább egy olyan osztály lesz, amelynek nem az első emeleten van a legtöbb órája. Az osztályok kiírásának a sorrendje tetszőleges! A megfelelő osztályokat írja ki egy `első.txt` állományba a mintának megfelelően.
-