

Készíts mozijegy foglaló konzolos alkalmazást!

A Marvelek című remekbe szabott Marvel film premier előtti vetítésére hatalmas az érdeklődés, nyilvánvalóan minden rajongó szeretne jegyet szerezni magának az alkalomra. Az egyik mozi üzemeltetője nagylelkűen fel is ajánl 90 darab ingyen jegyet a fanok számára, melyhez csak egy szimpla kérdőívet kell kitöltenie a résztvevőknek.

1) int[,] NezoterGen() 2p

Generáld le a terem nézőterét egy kétdimenziós mátrix segítségével (10 sor, 9 oszlop), majd töltsd fel random számokkal 17 és 40 között, melyek a nézők életkorát fogják jelenteni!

2) void Megjelenit(int[,] nezoter) 4p

Rajzold ki a konzolra a nézőteret a korábban legenerált életkorok megjelenítésével!
Jelöld piros színnel a kiskorúak életkorát!

3) double AtlagKor(int[,] nezoter) 5p

Határozd meg a teremben ülő nézők átlagéletkorát!

4) int Kiskoruak(int[,] nezoter) 4p

Határozd meg, hogy hány kiskorú rajongó van a teremben!

5) bool FiatalabbE(int[,] nezoter, int sor, int oszlop) 6p

Kérd be a felhasználótól egy sor és oszlop számát a következőképp:
„sorindex#oszlopindex”. Állapítsd meg, hogy a környezetében ülő nézők (bal szomszéd, jobb szomszéd, felette ülő szomszéd, alatta ülő szomszéd) átlagéletkoránál fiatalabb-e a bekért széken ülő! A szélső oszlopokban és sorokban nem kell működnie a függvénynek!

A Main függvényen belül kérjük be a stringet, és ott szerezzük meg belőle a sor-oszlopszámot, és a kész értékkel hívjuk meg a függvényt!

6) double[] SorokAtlaga(int[,] nezoter) 4p

Add vissza egy 1D-s double tömbben az egyes sorok átlagéletkorát!

7) string[] KiskoruakHelyei(int[,] nezoter) 7p

Gyűjtsd ki egy 1D-s string tömbbe minden kiskorú ülőhelyének sor és oszlopszámát!

A tömb egy eleme például a következőképp néz ki: sor: 9, oszlop: 13

8) void Csokkeno(double[] atlagok) 2p

Rendezd csökkenő sorrendben a 7-es feladatban meghatározott átlagéletkorokat, és írasd is ki őket a konzolra! A csökkenő sorrendbe rendezéshez a javított buborékredezés algoritmusát használd!

3.5. Algoritmus Javított buborékredezés

Bemenet: $x - \mathbf{T}$ tömb, $n - \text{egész}$ (tömb mérete); ahol \mathbf{T} összehasonlítható

Kimenet: $x - \mathbf{T}$ rendezett tömb

```
1: eljárás JAVÍTOTTBUBORÉKRENDEZÉS(címszerint  $x : \mathbf{T}$  tömb,  $n : \text{egész}$ )
2:    $i \leftarrow n$ 
3:   ciklus amíg  $i \geq 2$ 
4:      $idx \leftarrow 0$ 
5:     ciklus  $j \leftarrow 1$ -től  $(i - 1)$ -ig
6:       ha  $x[j] > x[j + 1]$  akkor
7:          $x[j] \leftrightarrow x[j + 1]$ 
8:          $idx \leftarrow j$ 
9:       elágazás vége
10:    ciklus vége
11:     $i \leftarrow idx$ 
12:  ciklus vége
13: eljárás vége
```

Felhasznált változók és függvények

- x : A rendezni kívánt tömb. Az x tömb elemeinek összehasonlíthatónak kell lennie. Az eljárás a tömböt helyben rendezi.
 - n : A paraméterként átadott tömb mérete.
 - idx : Az utolsó csere helyét tároljuk benne. Ha $x[j]$ -t és $x[j + 1]$ -et cseréljük, akkor az idx értéke j lesz.
-

9) Teszteld le a Main függvényben az elkészített metódusok

működését!
