**Megoldás sablon 1. C. Beadandó Házi feladat.**

Név: Golyha Gergő

Neptunkód: A7MMZ1

A képen szöveg, képernyőkép, Betűtípus, dokumentum látható

Automatikusan generált leírásFeldat: Legidősebb ember\*

1. Részteljesítés max: 50 pont

|  |  |
| --- | --- |
| Specifikáció | max: 20 pont |
| [Link](http://progalap.elte.hu/specifikacio/?data=H4sIAAAAAAAACpVPQWrDMBD8yqKTnSpGkivSivrQQqAQmluh4PggEoWItpKxXVMIueedfUlXMW6UQw69SLOzszO7e9LWZm23dq076x1R5MkocD%2FH45LCu28QLHxT8ixzFQWEUMC8h2949vgsdU1DWQCqkTn9SAawalZuYRVYtwluoZxvFfCHwgX8ijjBHk1R%2FPL4ltgihFBIBOO3U4wubZWZPp3kbMIF3EDCxUjvfKADl7ORc7pOU0JJZ9quJarck43uNB7kFIiQiDoVfoApmB5JzBlqgJ3HzcRYoVVo%2F1uM52AHd3D602Ayosa0Xx8dUfxA44VkvNCQwO%2FlbPA8%2BQ8wMh%2F3YCxq8fOEELEXv%2BJ1d2EVj8eBF1aSnVUyUsm%2Fk%2FMrJ1eHX2NrTT9fAgAA)  Be: n∈N, kor∈Kor[1..n], Kor = Ev x Ho x Nap, Ev = N, Ho = N, Nap = N  Ki: ind∈N  Ef: 1<=n  Uf: (ind,) = MAX(i=1..n, (2014-kor[i].ev)\*30\*12 + (12-kor[i].ho)\*30 + (30-kor[i].nap)) | Itt megalkotva, vagy másolva más környezetből vagy kézzel írt megoldás fényképe |
| Sablon + forrásmegjelölés | max: 5 pont |
| Maximumkiválasztás  programozási minták.pdf / 7. oldal ([canvas](https://canvas.elte.hu/courses/38943/files/2451951?module_item_id=587641)) | Lehet képernyőkép vagy másolat az előadás anyagából.  Forrásmegjelölés: a forrásfájl neve (publikálás útvonala) és oldalszám/dia sorszáma. |
| Visszavezetési táblázat (megfeleltetés) | max: 5 pont |
| maxind ~ ind  e..u ~ 1..n  f(i) ~ (2014-kor[i].ev)\*30\*12 + (12-kor[i].ho)\*30 + (30-kor[i].nap) | A specifikáció és a minta közötti kapcsolat leírása |
| Algoritmus | max: 20 pont |
|  | Itt megalkotva vagy képernyőkép vagy kézzel írt megoldás fényképe |

1. Részteljesítés max 50 pont

Módosítások max 10 pont

Itt kell dokumentálni, ha a beadott tervet a visszajelzés vagy a kódolás és tesztelés során kiderülő részletek miatt módosítani kell. A dokumentáció tartalmazza

|  |  |
| --- | --- |
| a teljes módosított részt (pl. specifikáció), kiemeléssel megjelölve a módosítást, | valamint a módosítás rövid indokát. Pl: visszajelzés alapján; biro 3. teszteset; |

(A módosítások jellegétől függően, lehet szövegdoboz nyilakkal vagy lábjegyzet is az indoklás.)

Maximális pontszámot ér az összes módosítás dokumentálása. Ha nem volt módosítás és a kód a korábban beadott tervet képezi le, akkor megkapható a maximális pontszám (nem a mennyiség, hanem a minőség számít.) Ebben az esetben is szerepeljen itt az első részteljesítés.

Kód forrása max 11 pont

|  |  |
| --- | --- |
| biro.elte.hu téma: **Maximumkiválasztás** feladat: **Legidősebb ember** beadás (feltöltés) sorszáma: **4** pontszám: **100** | Break, exit vagy try-catch használata esetén a 11 pontból legfeljebb 1 pont adható. |
| Ha volt: hibás tesztesetek sorszáma és a mellettük megjelenő hibaüzenetek | A hibaüzenetek és a kód függvényében, akár 80%-os eredmény is lehet maximális pontszámú |

Kód max 26 pont

2-2-2 pont: megjegyzésbe írt információk, külön választott deklarálás és kiírás.

8 pont a külön választott beolvasás/adatok eltárolása Ezen belül break, exit vagy try-catch használata esetén a 8 pontból 0 pont adható.

12 pont az algoritmus szabványos kódolása. Ezen belül break, exit vagy try-catch használata esetén a 26 pontból 0 pont adható.

using System;

// Golyha Gergő

// A7MMZ1

// golyhagergo@gmail.com

// Maximumkiválasztás: Legidősebb ember

namespace beadando1

{

internal class Program

{

struct Kor

{

public int ev;

public int ho;

public int nap;

}

static void Main(string[] args)

{

///////////////////////////// Deklarálás ///////////////////////////////

int n;

Kor[] kor;

int ind;

///////////////////////////// Beolvasás ////////////////////////////////

Console.Error.Write("n = ");

// Előfeltétel: 1 <= n

int.TryParse(Console.ReadLine(), out n);

kor = new Kor[n];

for (int i = 0; i < n; i++)

{

Console.Error.Write("{0}. adat (év hónap nap): ", i + 1);

string[] be = Console.ReadLine().Split(" ");

int.TryParse(be[0], out kor[i].ev);

int.TryParse(be[1], out kor[i].ho);

int.TryParse(be[2], out kor[i].nap);

}

//////////////////////////// Feldolgozás ///////////////////////////////

// A feladat leírása miatt a legnagyobb megadható dátum a 2014.12.30, valamit minden év 12 hónapos és minden hónap 30 napos.

// Számoljuk ki, hogy ki milyen idős ehhez a maximális időpillanathoz képest, tehát a dátumok különbségét.

// A számolásokat napokkal végezzük el, mert itt ez a legkisebb időegység.

// A különbségek közül kell kiválasztani a maximumot, azaz a legtöbb napos embert, azaz a legidősebbet.

// Alkalmazzuk a maximumkiválasztás sablonját!

int maxért = (2014 - kor[0].ev) \* 12 \* 30 + (12 - kor[0].ho) \* 30 + (30 - kor[0].nap);

// maxért segédváltozó: szükésges a maximumkiválasztáshoz de a feladat nem kéri mint kimenet

ind = 0;

// az indexelés 0-tól kezdődik

for (int i = 1; i < n; i++)

{

if ((2014 - kor[i].ev) \* 12 \* 30 + (12 - kor[i].ho) \* 30 + (30 - kor[i].nap) > maxért)

{

maxért = (2014 - kor[i].ev) \* 12 \* 30 + (12 - kor[i].ho) \* 30 + (30 - kor[i].nap);

ind = i;

}

}

//////////////////////////////// Kiírás ////////////////////////////////

Console.Error.Write("A legidősebb ember sorszáma: ");

// a 0-tól induló indexelés miatt 1-et hozzá kell adni a kapott sorszámhoz, így kapjuk meg a végső eredményt

Console.Write(ind + 1);

}

}

}

Teszt max 3 pont

A feladathoz adott 2 teszten túl még 3 teszt.

(A tervmódosítást alátámasztó tesztek, speciális esetek tesztjei.)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Saját 1. | Saját 2. | Saját 3 |
| Bemenet: | 1  2004 12 20 | 3  1989 5 3  2003 11 3  1999 10 8 | 5  2008 4 11  1973 8 30  1982 2 20  1922 3 5  1954 6 15 |
| Kimenet: | 1 | 1 | 4 |