

Neptun kód: **UJI470** Név: **Nagy Levente**
Beadás verziószáma: 1.
Később kiegészítéssel(!)
2. (javított Uf...)
3. (javított Algoritmus)

Feladat

Legidősebb dolgozó

Egy vállalat tárolja dolgozói életkorát és fizetését.

Írj programot, amely megadja a legidősebb dolgozó fizetését!

Bemenet

A standard bemenet első sorában az emberek száma van ($1 \leq N \leq 100$), a következő N sorban egy-egy ember kora ($1 \leq K \leq 100$) és a fizetése van ($0 \leq F \leq 2\ 000\ 000$).

Kimenet

A standard kimenet első sorába egyetlen egész számot kell írni, a legidősebb dolgozó fizetését (ha több megoldás van, akkor közülük a legkisebb sorszámú fizetését)!

Specifikáció

<https://progalap.elte.hu/specifikacio/?data=H4sIAAAACpWR32rCMBTGX%2BWQK4Ug-rbW2BitssCuZN2MwsL1bZT4JxltB04Rdumr7Dn2JnuS0cTY1urGepWefN853y9nj7JXNuNzPqM5lwIRdM8IJDF8H48wwRAKAIBEr-hdy1eqOpYpT%2BTU7nSSOM-JaoGsQQGsUzKBLCz5juUsl5N2KJ4ogQ3dcgGwrWm8oduVTPVfkMacwJotBF28yzg21Ye5ijCC74%2Bs%2Bi4DMC2LKAige%2FjB1fCUxiCLRNzyqNO0fxsvKjrLjr5f0pN4yCK2ZzOQyga6mvDaF4nhN4vHtpLQOVBJeWpZrXlt7F8BvtyoKoMAelzWjLgQijnGV5hsh0jxKaU0RQEhNwQhEK4yMw1Uz7YgngWPi8BeiptAdcFXSdiqB7RdCrdnC0lBRRMdNEVBE0UaEPRUL-kLAgjQTcMEYQwSln2ts4RsQ%2B4BtL7BcT2KzEGvuN0L3LWSD3f8hogXvUp3CapX-xXY7g3UXonqe3XUk%2Bcqg%2FUPVs%2F9g7U%2FqLM2ltqvovYbL9Hc%2BRVQuwT13DqoznSDMzr8ADjF0zJQBAAA>

Sablon

Maximumkválasztás sablon

Feladat

Adott az egész számok egy $[e..u]$ intervalluma és egy $f[e..u] \rightarrow H$ függvény. A H halmaz elemein értelmezett egy teljes rendezési reláció. Határozzuk meg, hogy az f függvény hol veszi fel az $[e..u]$ nem üres intervallumon a legnagyobb értéket, és mondjuk meg, mekkora ez a maximális érték!

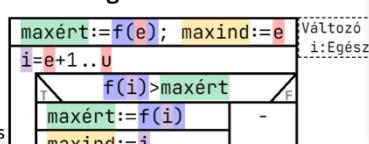
Specifikáció

Be: $e \in \mathbb{Z}$, $u \in \mathbb{Z}$
Ki: $\maxind \in \mathbb{Z}$, $\maxért \in H$
Ef: $e < u$
Uf: $\maxind \in [e..u]$ és
 $\forall i \in [e..u]: (f(\maxind) \geq f(i)) \text{ és } \maxind = f(\maxind)$

Rövidítve:

Uf: $(\maxind, \maxért) = \text{MAX}(i=e..u, f(i))$

Algoritmus



Megoldás sablon „C” beadandó 1. fázishoz (lecsérélendő a konkrét beadandó megnevezésére)

Visszavezetés

i	j
e..u	1..db
f(i)	max(dolgozok[j].kor)
maxindex	-
maxérték	-

Algoritmus

<https://progalap.elte.hu/stuki/?data=H4sIAAAAAAAACq1V70%2FbMBD9V6JDQpuwkClt0x9jEjAJygQMkMY2xAfHPrd-WUpSLm1B%2Bd%2BnpLGTQicNjX7J9eV8797z2XkJWAAAdJe29zu9XrbCcN%2Br03DsA0EJuYR05GAgZ4ICYEME%2BQWRY1olz-CDwTOcJaE96S8ui7is6AECDyxFbetFEyUEahhIlmRlwC4fEAaQ4e8Zao5AgE9UIL-LUI5HB4A6OsXt5GP4u3szjr9FjGD8BgfPW6PbXj2kM9zmpUxx9Y01N32zqL01YZnGK2glBiwsLA1DaBil-igTDj2DyZ%2BE5E97uxSV8gUj2hxYwEU7ZQWuAiOaho%2BQ9Ti3FwUKfTan2CY83GSxNFn6L0M%2BQEurwz-GrXOTr0MD6zJaLjwVi%2Fn6urq9lYlpWu%2BjqNrFH6La%2FOJSkouo4WyyujKNhUcBhvDQAXbiR0GlhoGamen1Eog-Mmj5U7U5WpOaE%2BiZ7EzK7KdvzAPrPjxb%2Fdhb9luXR32lkAgvj65VtnFT-WUKdk7F0dzz%2ByB92L%2BcoH0%2BFvMVjvg6vgdqAs36BpjlsjOyZf2y%2BCDHzhVDlywPbZDP5Afq32w6QyPUqb5ZG03Goa-XTbtSvDM5qe10Ohr%2BNnQ0qv7z%2Bdt0glaC3KnxG%2BjIPfAe5O4wK0fnLx5H54H%2FvmvqG6Gp1jXhLxnXiR8l14kH1jpp-jPDmTrQxD5DnBDKbzrg145RNi1l9Bs2mRcl5UxoIPLJUsSgbwnwAAqkxr6RmEzP%2FXqdVPAV6ikxgWkH5fTGui-UnLj8dK%2FR1sSUkppUBgg3jKYqi3pOQei2jHR7KMQrpPWRlxZLIXwv3KzCPG43FqZlpUhaNW1CrSiqcrjNxhGHLmlllGos0ptopyxT-fuHcs1PH5VtfhVmjjnnpSrOucNky2EoBV9h%2Ff7qLaU93u8UESK2RbfpsJ7%2FASkjzbzHmBwAA>

```
int db, dolgozok[db].kor, dolgozok[db].fizetes, maxindex = 0, maxertek = dolgozok[0].kor, legnagyobb
int i = 1; i < db; i++
    if (dolgozok[i].kor > maxertek)
        maxertek = dolgozok[i].kor
        maxindex = i
    legnagyobb = dolgozok[maxindex].fizetes
```

Kód (C#)

```
using System;
class Program
{
    /*
    Nagy Levente
    UJI470
    levente0517@gmail.com
    */

    public struct dolgozok
    {
        public int kor;
        public int fizetes;
        public dolgozok(int a, int b)
```

Megoldás sablon „C” beadandó 1. fázishoz (lecsérélendő a konkrét beadandó megnevezésére)

```
{  
    kor = a;  
    fizetes = b;  
}  
}  
  
static void Main(string[] args)  
{  
    int db = int.Parse(Console.ReadLine());  
  
    int maxkor = 0;  
  
    int maxindex = 0;  
  
    dolgozok[] dolgozok_korido = new dolgozok[db];  
  
    string[] temp = new string[db];  
  
    for (int i = 0; i < db; i++)  
    {  
        temp = Console.ReadLine().Split(' ');  
  
        dolgozok_korido[i] = new dolgozok(int.Parse(temp[0]), int.Parse(temp[1]));  
    }  
  
    for (int i = 0; i < db; i++)  
    {  
        if (dolgozok_korido[i].kor > maxkor)  
        {  
            maxkor = dolgozok_korido[i].kor;  
            maxindex = i;  
        }  
    }  
  
    Console.WriteLine(dolgozok_korido[maxindex].fizetes);  
}  
}
```

Bíró pontszám és képernyőkép

Result of the last submission

Total points: 100/

Test#	Point	Verdict	CPU time
1.1	3/3	Helyes	0.028 sec
2.1	3/3	Helyes	0.031 sec
3.1	3/3	Helyes	0.031 sec
4.1	3/3	Helyes	0.031 sec
5.1	3/3	Helyes	0.031 sec
6.1	3/3	Helyes	0.031 sec
7.1	3/3	Helyes	0.031 sec
8.1	3/3	Helyes	0.031 sec
9.1	4/4	Helyes	0.031 sec
10.1	4/4	Helyes	0.033 sec
11.1	4/4	Helyes	0.031 sec
12.1	4/4	Helyes	0.031 sec
13.1	4/4	Helyes	0.031 sec
14.1	4/4	Helyes	0.041 sec
15.1	4/4	Helyes	0.031 sec
16.1	4/4	Helyes	0.032 sec
17.1	4/4	Helyes	0.031 sec
18.1	4/4	Helyes	0.032 sec
19.1	4/4	Helyes	0.031 sec
20.1	4/4	Helyes	0.030 sec

21.1	4/4	Helyes	0.031 sec
22.1	4/4	Helyes	0.033 sec
23.1	4/4	Helyes	0.029 sec
24.1	4/4	Helyes	0.032 sec
25.1	4/4	Helyes	0.032 sec
26.1	4/4	Helyes	0.035 sec
27.1	4/4	Helyes	0.032 sec

Saját tesztfájlok

3	4	4
30 400000	18 983320	75 983320
23 200000	30 780700	69 78000
40 300000	47 450000	26 4600
300000	87 1500000	43 2000000
	1500000	983320