

Neptun kód: **MNDJ3P**
Beadás verziószáma: 1.

Név: Bartók Patrik Róbert

Feladat - Résztevők száma kategóriánként a kihíváson

Specifikáció

Résztevők száma kategóriánként a kihíváson

A „Kihívás napja”, 1991 óta hazánk legnépszerűbb lakossági sporteseménye. Magyarországon eddig összesen 1591 település vett részt a vetélkedésben. Tavaly a versenyben induló falvak és városok lakói egyetlen nap alatt 48 millió percnyi testmozgást végeztek.

Az eseményre nevezni kell a helységnévvel, a helységről tudjuk a lakosok számát, és hogy hányan vesznek részt az eseményen. N nevezés történt, amelyet időrendben jegyeztek föl. A feldolgozásban szerepet kap a helységek alábbi kategorizálása: I. kategória – a 700 főnél kisebb lélekszámúak, II. kategória – 700-1499 fő, III. kategória – 1500-2999 fő, IV. kategória – 3000-7999 fő, V. kategória – 8000-24 999 fő, VI. kategória – 25 000-69 999 fő, VII. kategória – 70 000 fő felettiek.

Készíts programot, amely megadja, hogy hányan vettek részt az eseményen az egyes kategóriákban!

Sablon

Másolás:

Specifikáció

Be: $e \in \mathbb{Z}$, $u \in \mathbb{Z}$

Ki: $y \in H[1..u-e+1]$

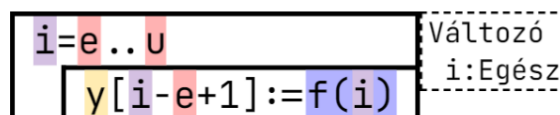
Ef: -

Uf: $\forall i \in [e..u] : (y[i-e+1] = f(i))$

Rövidítve:

Uf: $y = \text{MÁSOL}(i=e..u, f(i))$

Algoritmus



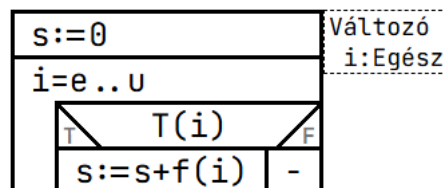
Feltételes összegzés:

Be: $e \in \mathbb{Z}$, $u \in \mathbb{Z}$

Ki: $s \in H$

Ef: -

Uf: $s = \text{SZUMMA}(i=e..u, f(i), T(i))$



Visszavezetés

Másolás:

y ~ osszegzes

e ~ 1

u ~ 7

f(i) ~ sum(i)

Összegzés:

s ~ sum

i ~ j

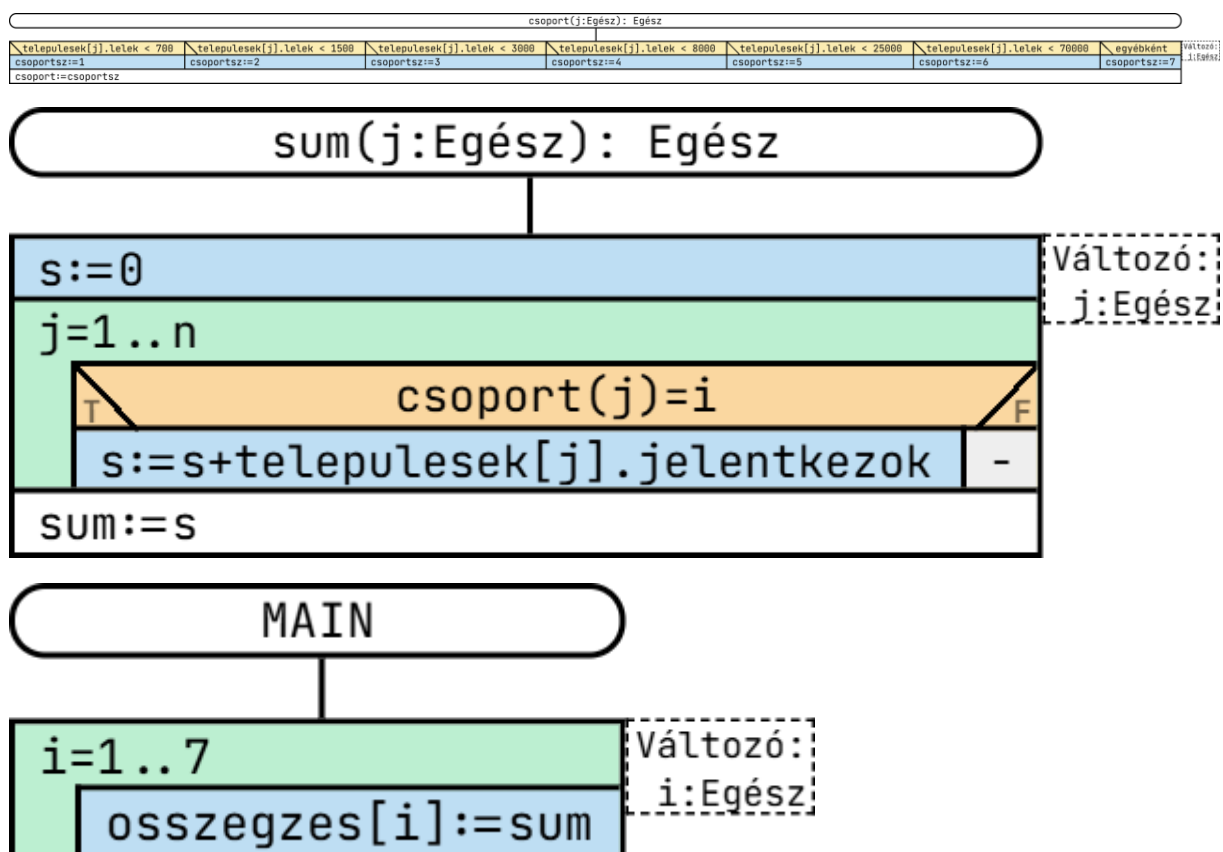
e ~ 1

u ~ n

f(i) ~ telepulesek[j].jelentkezo

T(i) ~ csoport(j)=i

Algoritmus



Kód (C#)

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Linq;
```

```
using System.Text;
```

```
using System.Threading.Tasks;
```

```
//Bartók Patrik Róbert - MNDJ3P - mndj3p@inf.elte.hu
```

```
namespace bigProject02
```

```
{
```

```
    internal class Program
```

```
    {
```

```
        class Tel
```

```
        {
```

```
            private string nev;
```

```
            private int lelek;
```

```
            private int jelentkezo;
```

```
            public string Nev
```

```
            {
```

```
                get { return nev; }
```

```
                set { nev = value; }
```

```
            }
```

```
            public int Lelek
```

```
            {
```

```
                get { return lelek; }
```

```
                set { lelek = value; }
```

```
            }
```

```
            public int Jelentkezo
```

```
            {
```

```
                get { return jelentkezo; }
```

```
        set { jelentkezo = value; }
    }
}

//Deklarálás
static int n;

static List<Tel> telepulesek = new List<Tel>();

static int[] osszegzes = new int[7];

static void read()
{
    n = int.Parse(Console.ReadLine());
    for (int i = 0; i < n; i++)
    {
        Tel temp = new Tel();
        temp.Nev = Console.ReadLine();
        temp.Lelek = int.Parse(Console.ReadLine());
        temp.Jelentkezo = int.Parse(Console.ReadLine());
        telepulesek.Add(temp);
    }
}

static int csoport(int j)
{
    int csoportsz;

    if (telepulesek[j].Lelek < 700)
    {
        csoportsz = 1;
    } else if (telepulesek[j].Lelek < 1500)
    {
        csoportsz = 2;
    }
}
```

```
    else if (telepulesek[j].Lelek < 3000)
    {
        csoportsz = 3;
    }
    else if (telepulesek[j].Lelek < 8000)
    {
        csoportsz = 4;
    }
    else if (telepulesek[j].Lelek < 25000)
    {
        csoportsz = 5;
    }
    else if (telepulesek[j].Lelek < 70000)
    {
        csoportsz = 6;
    }
    else
    {
        csoportsz = 7;
    }

    int csoport = csoportsz;
    return csoport;
}
```

```
static int sum(int i)
{
    int s = 0;

    for (int j = 0; j < n; j++)
    {
```

```
        if (csoport(j) == i+1)
        {
            s = s + telepulesek[j].Jelentkezo;
        }
    }

    int sum = s;
    return sum;
}

static void write()
{
    Console.WriteLine(osszegzes[0]);
    for (int i = 1; i < 7; i++)
    {
        Console.WriteLine(" " + osszegzes[i]);
    }
}

static void Main(string[] args)
{
    read();
    for (int i = 0; i < 7; i++)
    {
        osszegzes[i] = sum(i);
    }
    write();
}
}
```

Bíró pontszám és képernyőkép

100/100

Felhasználó: Bartók Patrik Róbert, Téma: Programozás 2. beadandó, Határidő: 2024-12-08 23:59:59

TÉMAVÁLTÁS

BEAD

EREDMÉNY

LETÖLTÉS

VISSZATÖLT

BEADOTTAK

FŐOLDAL

KILÉP/BELÉP

Utolsó beadás eredménye

Összpont: 100/

Teszt#	Pont	...Verdikt...	futási idő
1.1	10/10	Helyes	0.030 sec
2.1	10/10	Helyes	0.031 sec
3.1	10/10	Helyes	0.032 sec
4.1	10/10	Helyes	0.030 sec
5.1	10/10	Helyes	0.031 sec
6.1	10/10	Helyes	0.031 sec
7.1	10/10	Helyes	0.031 sec
8.1	10/10	Helyes	0.031 sec
9.1	10/10	Helyes	0.032 sec
10.1	10/10	Helyes	0.032 sec

Beadva: 2024-12-08 23:27:35.0

Saját tesztfájlok

testFile1_Be

3

Eger

1500

500

Pest

500

150

Nyíregy

70500

10000

TestFile1_Ki

150 0 500 0 0 0 10000

testFile2_Be

5

Eger

1500

500

Résztevők száma kategóriánként a kihíváson, Második Beadandó – C2

Pest

500

150

Nyíregy

70500

10000

Debrecen

155000

10000

Vác

7955

1250

TestFile2_Ki

150 0 500 1250 0 0 20000

testFile3_Be

4

Eger

1500

500

Pest

500

150

Nyíregy

70500

10000

Debrecen

155000

10000

TestFile3_Ki

150 0 500 0 0 0 20000