

„Programozás” komplex beadandó feladat

*Készítette: Ballai Péter Lóránt
Neptun-azonosító: EYSEED
E-mail: eyseed@inf.elte.hu*

*Kurzuskód: IT-18PROGEG
Gyakorlatvezető neve: Magyar Péter*

2026. január 09.

Tartalom

Felhasználói dokumentáció.....	3
Feladat.....	3
Futási környezet.....	3
Használat.....	3
A program indítása.....	3
A program használata billentyűzetről való bevitel esetén.....	3
A program használata fájlból való bevitel esetén.....	3
A program kimenete.....	4
Minta bemenet és kimenet.....	4
Hibalehetőségek.....	4
Fejlesztői dokumentáció.....	5
Feladat.....	5
Tervezés.....	5
Specifikáció.....	5
Visszavezetés.....	5
Algoritmus.....	6
Fejlesztői környezet.....	6
Forráskód.....	6
Megoldás.....	7
Függvénystruktúra.....	7
A kód.....	7
A kód (magas szintű függvényekkel).....	10
Tesztelés.....	11
Automatikus tesztek (Bíró).....	11
Automatikus tesztek (Bíró, magas szintű függvényekkel).....	12
Érvényes tesztesetek.....	13
Érvénytelen tesztesetek.....	13
Fejlesztési lehetőségek.....	14

Felhasználói dokumentáció

Feladat

Települések, ahol minden nap meleg volt

A meteorológiai intézet az ország N településére adott M napos időjárás előrejelzést... Készíts programot, amely megadja azokat a településeket, ahol minden hőmérséklet 0 fok feletti!

Futási környezet

IBM PC, dotnet futtatására alkalmas, 64-bites operációs rendszer (pl. Windows 11). Nem igényel egeret.

Használat

A program indítása

A programot a dotnet run paranccsal lehet futtatni.

A program használata billentyűzetről való bevétel esetén

A dotnet run parancs elindításával a program az adatokat a **billentyűzetről** olvassa be a következő sorrendben:

#	Adat	Magyarázat
1.	Települések száma (N)	Nemnegatív szám
2.	Napok száma (M)	Nemnegatív szám
3.	Városok (towns)	Tömb

A program kimenete

A program kiírja azoknak a helységeknek a darabszámát és a sorszámaikat, ahol csak olyan madarat láttunk, amelyik valahol máshol is előfordul.

Minta bemenet és kimenet

```
peti@fedora:/mnt/Projects/Uni/uni-semester-1/semester1/C#/assignments/town_temps_non_freezing$ dotnet run
3 5
10 15 12 10 10
-11 11 11 11 20
12 16 16 16 20
2 1 3
```

Hibalehetőségek

Az egyes bemeneti adatokat a fenti mintának megfelelően kell megadni. Hiba, ha bármelyik megadandó adat nem természetes szám. Hiba esetén a program azzal jelzi a hibát, hogy elszáll exceptionnel.

Minta futás hibás bemeneti adatok esetén:

```
peti@fedora: /mnt/Projects/Uni/uni-semester-1/semester1/C#/assignments/town_temps_non_freezing$ dotnet run
asd
Unhandled exception. System.FormatException: The input string 'asd' was not in a correct format.
   at System.Number.ThrowFormatException[TChar](ReadOnlySpan`1 value)
   at System.Int32.Parse(String s)
   at town_temps_non_freezing.Program.Main(String[] args) in /mnt/Projects/Uni/uni-semester-1/semester1/C#/assignments/town_temps_non_freezing/Program.cs:line 14
```

Fejlesztői dokumentáció

Feladat

Helységek csupa máshol is előforduló madárfajjal

Az ország néhány helységében madármegfigyelést végeztünk. Mindegyikben megadtuk, hogy milyen fajú madárból hányat láttunk.

Készíts programot, amely megadja azokat a helységeket, ahol csak olyan madarat láttunk, amelyik valahol máshol is előfordult!

Tervezés

Specifikáció

In: $n \in \mathbb{N}$, $m \in \mathbb{N}$, $\text{towns} \in \mathbb{R}[1..n*m]$

Out: $t \in \mathbb{N}$, $\text{ids} \in \mathbb{R}[1..t]$

Pre: $n \geq 1$ and $n \leq 1000$ and

$m \geq 1$ and $m \leq 1000$ and

$\forall i \in [1..n*m]: (\text{towns}[i/n + i\%m] \geq -50)$ and

$\forall i \in [1..n*m]: (\text{towns}[i/n + i\%m] \leq 50)$

Post: $(t, \text{ids}) = \text{FILTER}(i=1..n, \forall j \in [1..m]: (\text{towns}[(i-1)*m + j] > 0), i)$

Visszavezetés

Kiválogatás

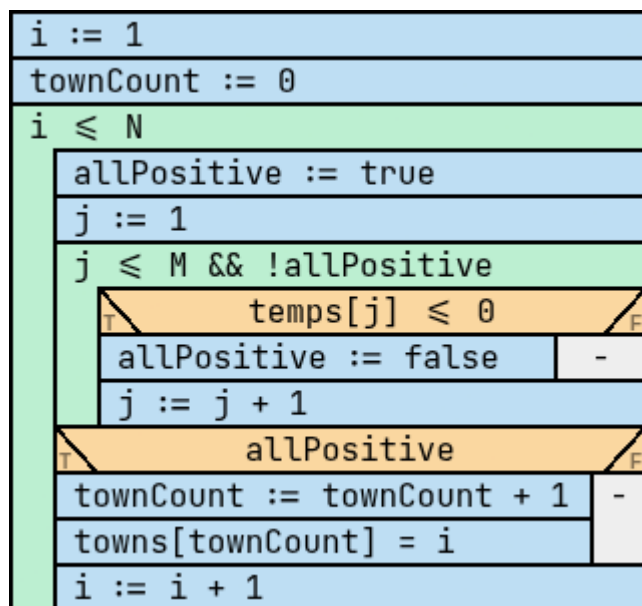
$y \sim \text{ids}$

$e..u \sim 1..N*M$

$T(i) \sim \forall j \in [1..m]: (\text{towns}[(i-1)*m + j] > 0)$

$f(i) \sim i$

Algoritmus



Fejlesztői környezet

IBM PC, dotnet futtatására alkalmas operációs rendszer (pl. Windows 11 Home). Dotnet-et futtatni képes terminál, vagy más futtatási környezet.

Forráskód

A teljes fejlesztői anyag az `town_temps_non_freezing` nevű könyvtárban található meg. A fejlesztés során használt könyvtár-struktúra:

Állomány	Magyarázat
A1B2C3/obj/	mappa fordításhoz szükséges kódokkal
A1B2C3/Program.cs	C# forráskód
A1B2C3/documentation.pdf	dokumentációk (ez a fájl)

Megoldás

A kód

A `Program.cs` fájl tartalma:

```
/*  
    Készítette: Ballai Péter Lóránt  
    Neptun: EYSEED  
    E-mail: eyseed@inf.elte.hu  
    Feladat: Madármegfigyelés/ Helységek csupa máshol is előforduló madárfajjal  
*/
```

```
using System;  
using System.Collections.Generic;  
  
namespace town_temps_non_freezing  
{  
    class Program  
    {  
        static void Main(string[] args)  
        {  
            string line = Console.ReadLine();  
            if (string.IsNullOrEmpty(line)) return;  
  
            string[] parts = line.Split(' ');  
            int N = int.Parse(parts[0]);  
            int M = int.Parse(parts[1]);  
            List<int> goodTowns = new List<int>();  
  
            for (int i = 0; i < N; i++)  
            {  
                string rowInput = Console.ReadLine();  
                string[] temps = rowInput.Split(' ');
```

```

bool allPositive = true;

for (int j = 0; j < M; j++)
{
    int temp = int.Parse(temps[j]);
    if (temp <= 0)
    {
        allPositive = false;
        break;
    }
}

if (allPositive)
{
    goodTowns.Add(i + 1);
}

Console.Write($"{goodTowns.Count} ");

for(int i = 0; i < goodTowns.Count; i++)
{
    Console.Write($"{goodTowns[i]} ");
}
System.Console.WriteLine();
}
}
}

```

A kód (magas szintű függvényekkel)

A Program.cs fájl tartalma:

```

using System;
using System.Collections.Generic;
using System.Linq;

namespace town_temps_linq
{
    class Program
    {
        static void Main(string[] args)
        {
            string line = Console.ReadLine();
            if (string.IsNullOrEmpty(line)) return;
            string[] parts = line.Split(' ');
            int N = int.Parse(parts[0]);
            List<List<int>>> adatok = new List<List<int>>>();

```

```
for (int i = 0; i < N; i++)
{
    adatok.Add(Console.ReadLine().Split(' ').Select(int.Parse).ToList());
}
```

```
var megoldas = Enumerable.Range(0, N)
    .Where(i => adatok[i].All(homerseklet => homerseklet > 0))
    .Select(i => i + 1)
    .ToList();
```

Összpont: 100/

```
Console.WriteLine($"{megoldas.Count} ");
foreach (var varos in megoldas)
{
    Console.WriteLine($"{varos} ");
}
}
```

Teszt#	Pont	...Verdikt...	futási idő
1.1	4/4	Helyes	0.030 sec
2.1	4/4	Helyes	0.032 sec
3.1	4/4	Helyes	0.032 sec
4.1	4/4	Helyes	0.032 sec
5.1	4/4	Helyes	0.032 sec
6.1	5/5	Helyes	0.031 sec
7.1	5/5	Helyes	0.032 sec
8.1	5/5	Helyes	0.031 sec
9.1	5/5	Helyes	0.032 sec
10.1	5/5	Helyes	0.033 sec
11.1	5/5	Helyes	0.032 sec
12.1	5/5	Helyes	0.032 sec
13.1	5/5	Helyes	0.034 sec
14.1	5/5	Helyes	0.033 sec
15.1	5/5	Helyes	0.034 sec
16.1	6/6	Helyes	0.051 sec
17.1	6/6	Helyes	0.055 sec
18.1	6/6	Helyes	0.051 sec
19.1	6/6	Helyes	0.081 sec
20.1	6/6	Helyes	0.204 sec

Beadva: 2025-12-03 12:21:58.0

Tesztelés

Automatikus tesztek (Bíró)¹

¹ Példaképpen egy Mesteres képernyőkép van itt megadva. A végső beadandóban a Bíróból kell a képet kivágni, méghozzá az Eredmény menüpontból kiválasztva a megfelelő „Próbát”.

Automatikus tesztek (Bíró, magas szintű függvényekkel)

Összpont: 100/

Teszt#	Pont	...Verdikt...	futási idő
1.1	4/4	Helyes	0.030 sec
2.1	4/4	Helyes	0.032 sec
3.1	4/4	Helyes	0.032 sec
4.1	4/4	Helyes	0.032 sec
5.1	4/4	Helyes	0.032 sec
6.1	5/5	Helyes	0.031 sec
7.1	5/5	Helyes	0.032 sec
8.1	5/5	Helyes	0.031 sec
9.1	5/5	Helyes	0.032 sec
10.1	5/5	Helyes	0.033 sec
11.1	5/5	Helyes	0.032 sec
12.1	5/5	Helyes	0.032 sec
13.1	5/5	Helyes	0.034 sec
14.1	5/5	Helyes	0.033 sec
15.1	5/5	Helyes	0.034 sec
16.1	6/6	Helyes	0.051 sec
17.1	6/6	Helyes	0.055 sec
18.1	6/6	Helyes	0.051 sec
19.1	6/6	Helyes	0.081 sec
20.1	6/6	Helyes	0.204 sec

Beadva: 2025-12-03 12:21:58.0

Érvényes tesztesetek

1.

Bemenet
0 0
Kimenet
0

2.

Bemenet
1 1 1
Kimenet
1 1

3.

Bemenet
3 5 10 15 12 10 10 -11 11 11 11 20 12 16 16 16 20
Kimenet
2 1 3

Érvénytelen tesztesetek

Billentyűzetes bevétel esetén

4. *teszteset*

Bemenet – szöveges adat
N = asd
Kimenet
Unhandled exception. System.FormatException: The input string 'asd' was not in a correct format. at System.Number.ThrowFormatException[TChar](ReadOnlySpan`1 value) at System.Int32.Parse(String s) at town_temps_non_freezing.Program.Main(String[] args) in /mnt/Projects/Uni/uni-semester-1/semester1/C#/assignments/town_temps_non_freezing/

Program.cs:line 14

5. *teszteset*

Bemenet – <i>Negatív szám</i>
N = -1
Kimenet
Unhandled exception. System.IndexOutOfRangeException: Index was outside the bounds of the array. at town_temps_non_freezing.Program.Main(String[] args) in /mnt/Projects/Uni/uni-semester-1/semester1/C#/assignments/town_temps_non_freezing/ Program.cs:line 15

Fejlesztési lehetőségek

1. Exception-ök kezelése