**„Programozás”  
komplex beadandó feladat**

Készítette: *Pőcze Máté*Neptun-azonosító: *EBYPPB*E-mail: ebbyppb@inf.elte.hu

Kurzuskód: IT-18PROGEGGyakorlatvezető neve: *Éles Júlia*

2025. január 11.

Tartalom

[Felhasználói dokumentáció 3](#_Toc186985036)

[Feladat 3](#_Toc186985037)

[Futási környezet 3](#_Toc186985038)

[Használat 3](#_Toc186985039)

[A program indítása 3](#_Toc186985040)

[A program használata billentyűzetről való bevitel esetén 3](#_Toc186985041)

[A program használata fájlból való bevitel esetén 3](#_Toc186985042)

[A program kimenete 4](#_Toc186985043)

[Minta bemenet és kimenet 4](#_Toc186985044)

[Hibalehetőségek 4](#_Toc186985045)

[Fejlesztői dokumentáció 5](#_Toc186985046)

[Feladat 5](#_Toc186985047)

[Specifikáció 5](#_Toc186985048)

[Visszavezetés 5](#_Toc186985049)

[Algoritmus 5](#_Toc186985050)

[Fejlesztői környezet 5](#_Toc186985051)

[Forráskód 6](#_Toc186985052)

[Megoldás 6](#_Toc186985053)

[Függvénystruktúra 6](#_Toc186985054)

[A kód 7](#_Toc186985055)

[A kód (magas szintű függvényekkel) 10](#_Toc186985056)

[Tesztelés 13](#_Toc186985057)

[Automatikus tesztek (Bíró) 13](#_Toc186985058)

[13](#_Toc186985059)

[Automatikus tesztek (Bíró, magas szintű függvényekkel) 14](#_Toc186985060)

[Érvényes tesztesetek 15](#_Toc186985061)

[Érvénytelen tesztesetek 16](#_Toc186985062)

[Fejlesztési lehetőségek 16](#_Toc186985063)

2. Felhasználói dokumentáció
   1. Feladat

**Minden településen melegebb van**

A meteorológiai intézet az ország N településére adott M napos időjárás előrejelzést, az adott településen az adott napra várt legmagasabb hőmérsékletet.

A program megadja azokat a napokat, amikor minden településen melegebb lesz, mint az előző nap!

* 1. Futási környezet

Exe futtatására alkalmas PC, 64-bites operációs rendszer (pl. Windows 11). A program más módon nem futtatható (pl. mobilról, vagy böngészőben). A program használatához billentyűzet szükséges.

* 1. Használat
     1. A program indítása

A program az EBYPPB\beadando\_alacsony\bin\Debug\net9.0\beadando\_alacsony.exe néven található a tömörített állományban.

* + 1. A program használata billentyűzetről való bevitel esetén

Az beadando\_alacsony.exe fájl elindításával a program az adatokat a **billentyűzet**ről olvassa be a következő sorrendben:

|  |  |  |
| --- | --- | --- |
| # | Adat | Magyarázat |
| **1.** | *Települásek száma (n)* | Nemnegatív szám (1 és 1000 között) |
| **2.** | *Napok száma (m)* | Nemnegatív szám (1 és 1000 között) |
| **3.** | *1. településen 1. nap hőfoka* | Egész szám innentől (-50 és 50 között) |
| **4.** | *1. településen 2. nap hőfoka* |  |
| **…** | *…* |  |
|  | *1. településen m. nap hőfoka* |  |
|  | *2. településen 1. nap hőfoka* |  |
|  |  |  |
|  | *n. településen m. nap hőfoka* |  |

* + 1. A program használata fájlból való bevitel esetén

Lehetőségünk van az adatokat **fájl**ban is megadni. Ekkor a programot *parancssorban* a következőképpen kell indítani, feltételezve, hogy a bemeneti fájlok mellette helyezkednek el:

beadando\_alacsony.exe < be1.txt

A fájl felépítésének a következő formai követelményei vannak. A fájl első sorában a települések száma (n) és a hőfokok száma (m) van. A következő n sor mindegyikében m darab szám szerepel, közülük az i-edik sorban a j-edik szám az i-edik településen a j-edik napon jósolt hőmérséklet. Például:

3 4

10 9 12 14

-20 -15 0 10

20 25 24 30

* + 1. A program kimenete

A program kiírja azt a számot, ahányszor minden településen melegebb lesz, mint az előző nap, és ezt követően felsorolja ezeknek a napoknak a sorszámát növekvő sorrendben.

* + 1. Minta bemenet és kimenet

A képen szöveg, képernyőkép, Betűtípus, tervezés látható

Automatikusan generált leírás

* + 1. Hibalehetőségek

Az egyes bemeneti adatokat a fenti mintának megfelelően kell megadni.  
Hiba, ha a települések száma vagy napok száma nem 1 és 1000 közé eső egész szám, illetve, ha a hőmérsékletek nem -50 és 50 közé eső egész számok.  
Hiba esetén a program azzal jelzi a hibát, hogy újra kérdezi azt.

Mintafutás hibás bemeneti adatok esetén:

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

1. Fejlesztői dokumentáció
   1. Feladat

**Minden településen melegebb van**

A meteorológiai intézet az ország N településére adott M napos időjárás előrejelzést, az adott településen az adott napra várt legmagasabb hőmérsékletet.

A program megadja azokat a napokat, amikor minden településen melegebb lesz, mint az előző nap!

* 1. Tervezés
     1. [Specifikáció](https://progalap.elte.hu/specifikacio/?data=H4sIAAAAAAAACpWQ3U7CMBTHX%2BWkV1tSyDoGaBNINH6EgHij3oxdDClSxgph0xtConf4Hr6Jb8KTeLqtDA0SbdKe0%2FPR%2F69nRZKFeJRj%2BRimcq4IJ%2BeCg4LtZgN9CrFxJrnjs2pVUcAzDgbLgepKDpGpUeFiHpV1UVZx9cIhFjPxJIZDDn2otKFHdQJwmYQ1taEFN53%2BhSVbucTElxSmAbQLDyrAAls3Xo45MNi%2Bf2hMPJnjOBCqURGMvwWN0nbzKpEs4w%2B4hddpcY0DDpaTNRlN84Cd6d2jnhUV39Oc3c7D51vv9vrsDmldfIGWH5E2BWkTSlKRpAnh%2FoqMwjTEuSoONf1czKGu7YTnbBXwmYMjreNu4Na%2BE5Q5hne9a7l193Nu0dPMbZGLcD7aZsA7lRpCqTAWiILeUiTPs5RwZ03%2FTdg8QugdITzZJ3QPEBrH%2BwuqV6B6v6F6WvIHpmtQDiDqfKOMI%2B0pBW%2F318PM7nFmtg7WX9fx4ztjAwAA)

Be: n ∈ N, m ∈ N, h ∈ N[1..n, 1..m]

Ki: k ∈ N, napok ∈ N[1..k]

Fv: melegebb: N -> L,

melegebb(j) = MIND(i=1..n, h[i, j] > h[i, j - 1])

Ef: 1 ≤ n ≤ 1000 and 1 ≤ m ≤ 1000 and

∀i∈[1..n]:(∀j∈[1..m]: (0 ≤ h[i, j] ≤ 1000))

Uf: (k, napok) = KIVÁLOGAT(i=2..m, melegebb(i), i)

* + 1. Visszavezetés

*Mind eldöntés (melegebb)*

e..u ~ 1..n

T(i) ~ h[i, j] > h[i, j - 1]

*Kiválogatás (fő)*

e..u ~ 2..m

T(i) ~ melegebb(i)

f(i) ~ i

* + 1. A képen szöveg, képernyőkép, Betűtípus, szám látható

       Automatikusan generált leírás[Algoritmus](https://progalap.elte.hu/stuki/?data=H4sIAAAAAAAACq1Wa2%2FaMBT9K5EnVavGqgDlEbpOKtu0PgBVqqAtiA9%2BhbgEm4ZQYBX%2FfXbAF9PSqaizkHI5ts%2B5D%2Fsmz0gwVEP%2BkV8Oir7%2BlQvFYr5ULVdQDkXqiScXel5O4ziHJjzmNOVsg0jF%2BATVntGPoeIRz0fGzggByKExTrhMN5siwRiXqBbieMJzKF2Mud4w4Y9TLinXG2gkYqb3XDDN3UMNJaPm2bShZ%2FisURrOOo%2Bov8yh%2B%2FoVlotwBpoAfFyzkN43qq12Wc%2B0rn%2FO2%2Byyqk3RusN3j8N8Jg9uWXnHz428m4g3nEhxykd6uV6R8nmqoaFXO%2FX8byT5jrRQk4jb9u2kDkIAbAlBbvaP9rrT7ZL0zzyLC3isnEO8T1wzrZFpKclEKpRegMRp4ehopEGi2OJm7c%2FFVkzaAxW2updTHIAHAGx5AF7vH%2FD5WZ4HwcLXM93f5IaQuJHFHjQ79askLIMyAFvK5zM275bv6ZvKUqmxiQQWWj5n5%2F%2BJBBw07gOPlXOIHTnn%2FOyWE%2BGLuo30zR9wQj6LQ3NKkymvJ1jSaKuETpUyuh1LIHyTG1sDyM2mKI6zDuu77w8j5gIx8iVv7xBU2YoB8GExicdq2GOkbzSFFYQGYgWdjuIIOi3r3YLCCEFks%2FkoEGfHBIQA2BJyuti%2BR6wq2w9h%2BRfNjhjwWDmHeJ%2B4dnYH7yBOT0496WHJvKgnct5D3zsYpCfrP1%2Fz%2FSzsHf0DotY%2BgsPWRwC2fHQStV%2FuxeZcwRvBSgHw4TLbO5cprrIgV7Jad6LvIE3VIMEjU6NnfQpHhqeJhdQUTzgRmMTm3YxEraWRRKlXfXsSqVlns9Jc6xV4zjHjyQpZ5oB80wVqXuvQq3mNdTX%2BIedE%2FpJ5pwPLvjkTsUqyz4pVKnroUxj6emiST%2BYZmv6kMQoY8UtghZlV8Ys%2BzizKcVitoP4qs3VMh4NETSVbE5MCKZhl5mmJObUYr1BsrTCz2DH1ecHQma%2BfV3RUj0xVD0sXFizGQ0bfonNq%2BorVjHXM7%2BIPwyBYzfp%2BlQYlY3HOj1nZzchy%2BRcYPubn%2FwkAAA%3D%3D)
  1. Fejlesztői környezet

Exe futtatására alkalmas PC, és operációs rendszer (Windows 11). JetBrains Rider (2024.3.3), (net9.0 framework, konzol applikáció (C#)) fejlesztői környezet.

* 1. Forráskód

A teljes fejlesztői anyag –kicsomagolás után– az EBYPPB nevű könyvtárban található meg. A fej­lesztés során használt könyvtár-struktúra:

|  |  |
| --- | --- |
| Állomány | Magyarázat |
| EBYPPB\beadando\_alacsony\bin\Release\net9.0\beadando\_alacsony.exe | futtatható kód (a futtatáshoz szükséges fájlokkal) – alacsony megvalósítás |
| EBYPPB\beadando\_alacsony\obj\ | mappa fordításhoz szükséges kódokkal |
|  |  |
| EBYPPB\beadando\_alacsony\EBYPPB\_alacsony.cs | C# forráskód – alacsony megvalósítás |
|  |  |
| EBYPPB\beadando\_magas\bin\Release\net9.0\beadando\_magas.exe | futtatható kód (a futtatáshoz szükséges fájlokkal) – magas megvalósítás |
|  |  |
| EBYPPB\beadando\_magas\obj\ | mappa fordításhoz szükséges kódokkal |
|  |  |
| EBYPPB\beadando\_magas\EBYPPB\_magas.cs | C# forráskód – magas megvalósítás |
|  |  |
| EBYPPB\tesztek\teszt1.txt | teszt-bemeneti fájl1 |
| EBYPPB\tesztek\ki1.txt | teszt-kimeneti fájl1 |
| … |  |
| EBYPPB\tesztek\teszt8.txt | teszt-bemeneti fájl8 |
| EBYPPB\tesztek\ki8.txt | teszt-kimeneti fájl8 |
|  |  |
| EBYPPB\doksi\EBYPPB.docx | dokumentációk (ez a fájl) |

Megoldás

* + 1. Függvénystruktúra

* + 1. A kód

A EBYPPB\_alacsony.cs fájl tartalma:

*//#define BIRO*using System;  
using System.Collections.Generic;  
  
namespace beadando\_komplex  
{  
 internal class EBYPPB\_alacsony  
 {  
 static int n, m;  
 static int[,] h;  
  
 static void beolvas()  
 {  
 string bemenet;  
 string[] darabok;  
   
#if BIRO  
 bemenet = Console.ReadLine();  
 darabok = bemenet.Split(' ', StringSplitOptions.RemoveEmptyEntries);  
  
 n = int.Parse(darabok[0]);  
 m = int.Parse(darabok[1]);  
   
 h = new int[n + 1, m + 1];  
  
 for (int i = 1; i <= n; i++)  
 {  
 bemenet = Console.ReadLine();  
 darabok = bemenet.Split(' ', StringSplitOptions.RemoveEmptyEntries);  
  
 for (int j = 1; j <= m; j++)  
 {  
 h[i, j] = int.Parse(darabok[j - 1]);  
 }  
 }  
   
#else  
  
 bool hiba;  
  
 do  
 {  
 Console.ResetColor();  
 Console.Write("Települések száma: ");  
   
 bemenet = Console.ReadLine();  
   
 hiba = !int.TryParse(bemenet, out n) || n < 1 || n > 1000;  
  
 if (hiba)  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkRed**;  
 Console.WriteLine("Helytelen formátum! (1-1000)");  
 }  
 } while(hiba);  
   
 do  
 {  
 Console.ResetColor();  
 Console.Write("Napok száma: ");  
   
 bemenet = Console.ReadLine();  
   
 hiba = !int.TryParse(bemenet, out m) || m < 1 || m > 1000;  
  
 if (hiba)  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkRed**;  
 Console.WriteLine("Helytelen formátum! (1-1000)");  
 }  
 } while(hiba);  
   
 h = new int[n + 1, m + 1];  
   
 for (int i = 1; i <= n; i++)  
 {  
 for (int j = 1; j <= m; j++)  
 {  
 do  
 {  
 Console.ResetColor();  
 Console.Write($"{i}. település {j}. napi hőmérséklete: ");  
   
 bemenet = Console.ReadLine();  
   
 hiba = !int.TryParse(bemenet, out h[i, j]) || h[i, j] < -50 || h[i, j] > 50;  
   
 if (hiba)  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkRed**;  
 Console.WriteLine("Helytelen formátum! (-50-50)");  
 }  
 } while (hiba);  
 }  
 }  
#endif  
 }  
  
 static bool melegebb(int j)  
 {  
 int i = 1;  
 while (i <= n && h[i, j] > h[i, j - 1])  
 {  
 i++;  
 }  
  
 return i > n;  
 }  
  
 static (int Darab, List<int> Napok) feldolgozas()  
 {  
 int k = 0;  
 List<int> napok = new List<int>();  
  
 for (int i = 2; i <= m; i++)  
 {  
 if (melegebb(i))  
 {  
 k++;  
 napok.Add(i);  
 }  
 }  
   
 return (k, napok);  
 }  
  
 static void kiir((int Darab, List<int> Napok) t)  
 {  
#if BIRO  
 Console.WriteLine(t.Darab + " " + string.Join(" ", t.Napok));  
#else  
 if (t.Darab > 1)  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkGreen**;  
 Console.WriteLine($"{t.Darab} napon volt melegebb, ezeken a napokon: {string.Join(", ", t.Napok)}");  
 }  
 else if (t.Darab == 1)  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkGreen**;  
 Console.WriteLine($"{t.Darab} napon volt melegebb, ezen a napon: {string.Join(", ", t.Napok)}");   
 }  
 else  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkGreen**;  
 Console.WriteLine("Nem volt egyszer sem, hogy mindenhol melegebb volt, mint az előző nap!");  
 }  
 Console.BackgroundColor = ConsoleColor.**White**;  
 Console.ForegroundColor = ConsoleColor.**Black**;  
 Console.Write("Kérem, nyomjon ENTER-t a folytatáshoz!");  
 Console.ResetColor();  
 Console.ReadLine(); *// hogy ne zárja be rögtön a program magát*#endif  
 }  
   
 static void Main(string[] args)  
 {  
 beolvas();  
 kiir(feldolgozas());  
 }  
 }  
}

* + 1. A kód (magas szintű függvényekkel)

A EBYPPB\_magas.cs fájl tartalma:

*//#define BIRO*using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Numerics;  
using System.Text;  
using System.Threading.Tasks;  
using magas\_szintű\_mintamegvalósítások;  
  
namespace beadando\_k\_h  
{  
 internal class EBYPPB\_magas  
 {  
 static int n, m;  
 static int[,] h;  
   
 static void beolvas()  
 {  
 string bemenet;  
 string[] darabok;  
   
#if BIRO  
 bemenet = Console.ReadLine();  
 darabok = bemenet.Split(' ', StringSplitOptions.RemoveEmptyEntries);  
  
 n = int.Parse(darabok[0]);  
 m = int.Parse(darabok[1]);  
   
 h = new int[n + 1, m + 1];  
  
 for (int i = 1; i <= n; i++)  
 {  
 bemenet = Console.ReadLine();  
 darabok = bemenet.Split(' ', StringSplitOptions.RemoveEmptyEntries);  
  
 for (int j = 1; j <= m; j++)  
 {  
 h[i, j] = int.Parse(darabok[j - 1]);  
 }  
 }  
#else  
 bool hiba;  
  
 do  
 {  
 Console.ResetColor();  
 Console.Write("Települések száma: ");  
   
 bemenet = Console.ReadLine();  
   
 hiba = !int.TryParse(bemenet, out n) || n < 1 || n > 1000;  
  
 if (hiba)  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkRed**;  
 Console.WriteLine("Helytelen formátum! (1-1000)");  
 }  
 } while(hiba);  
   
 do  
 {  
 Console.ResetColor();  
 Console.Write("Napok száma: ");  
   
 bemenet = Console.ReadLine();  
   
 hiba = !int.TryParse(bemenet, out m) || m < 1 || m > 1000;  
  
 if (hiba)  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkRed**;  
 Console.WriteLine("Helytelen formátum! (1-1000)");  
 }  
 } while(hiba);  
   
 h = new int[n + 1, m + 1];  
   
 for (int i = 1; i <= n; i++)  
 {  
 for (int j = 1; j <= m; j++)  
 {  
 do  
 {  
 Console.ResetColor();  
 Console.Write($"{i}. település {j}. napi hőmérséklete: ");  
   
 bemenet = Console.ReadLine();  
   
 hiba = !int.TryParse(bemenet, out h[i, j]) || h[i, j] < -50 || h[i, j] > 50;  
   
 if (hiba)  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkRed**;  
 Console.WriteLine("Helytelen formátum! (-50-50)");  
 }  
 } while (hiba);  
 }  
 }  
#endif  
 }  
  
 static int[] feldolgozas()  
 {  
 int[] napok = Mintak.Kivalogat(2, m, j => Mintak.Mind(1, n, i => h[i, j] > h[i, j - 1]), j => j);  
   
 return napok;  
 }  
  
 static void kiir(int[] napok)  
 {  
#if BIRO  
 Console.WriteLine(napok.Length + " " + string.Join(" ", napok));  
#else  
 if (napok.Length > 1)  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkGreen**;  
 Console.WriteLine($"{napok.Length} napon volt melegebb, ezeken a napokon: {string.Join(", ", napok)}");  
 }  
 else if (napok.Length == 1)  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkGreen**;  
 Console.WriteLine($"{napok.Length} napon volt melegebb, ezen a napon: {string.Join(", ", napok)}");   
 }  
 else  
 {  
 Console.ForegroundColor = ConsoleColor.**DarkGreen**;  
 Console.WriteLine("Nem volt egyszer sem, hogy mindenhol melegebb volt, mint az előző nap!");  
 }  
 Console.BackgroundColor = ConsoleColor.**White**;  
 Console.ForegroundColor = ConsoleColor.**Black**;  
 Console.Write("Kérem, nyomjon ENTER-t a folytatáshoz!");  
 Console.ResetColor();  
 Console.ReadLine(); *// hogy ne zárja be rögtön a program magát*#endif  
 }  
   
 static void Main(string[] args)  
 {  
 beolvas();  
 kiir(feldolgozas());  
 }  
 }  
}

* 1. Tesztelés
     1. Automatikus tesztek (Bíró)
     2. A képen szöveg, képernyőkép, szám, Betűtípus látható

        Automatikusan generált leírás
     3. Automatikus tesztek (Bíró, magas szintű függvényekkel)

A képen szöveg, képernyőkép, szám, sor látható

Automatikusan generált leírás

* + 1. Érvényes tesztesetek
       1. teszteset: teszt1.txt

|  |
| --- |
| Bemenet – *3 település, 3 nap – 1 megoldás* |
| 3 3  10 11 12  9 10 3  10 15 30 |
| Kimenet |
| 1 2 |

* + - 1. teszteset: teszt2.txt

|  |
| --- |
| Bemenet – *4 település, 4 nap – több megoldás* |
| 4 4  -10 -8 -5 -11  10 16 17 20  -20 -19 -14 -10  30 34 36 40 |
| Kimenet |
| 2 2 3 |

* + - 1. teszteset: teszt3.txt

|  |
| --- |
| Bemenet – *3 település, 4 nap – 0 megoldás* |
| 3 4  10 9 8 7  20 17 14 11  30 31 32 36 |
| Kimenet |
| 0 |

* + - 1. teszteset: teszt4.txt

|  |
| --- |
| Bemenet – *4 település, 5 nap – 1 megoldás* |
| 4 5  10 15 14 16 17  20 21 22 18 17  -10 -5 -2 -7 -10  30 35 32 31 29 |
| Kimenet |
| 1 2 |

* + - 1. teszteset: teszt5.txt

|  |
| --- |
| Bemenet – *4 település, 3 nap – több megoldás* |
| 4 3  10 11 13  20 21 23  30 31 33  40 41 43 |
| Kimenet |
| 2 2 3 |

* + 1. Érvénytelen tesztesetek

Billentyűzetes bevitel esetén

* + - 1. teszteset: teszt6.txt

|  |
| --- |
| Bemenet – *szöveges adat* |
| 1. Települések száma: három |
| Kimenet |
| 1. Helytelen formátum! (1-1000) 2. Települések száma: |

* + - 1. teszteset: teszt7.txt

|  |
| --- |
| Bemenet – *előfeltételnek nem megfelelő bemenet* |
| Települések száma: 1001 |
| Kimenet |
| 1. Helytelen formátum! (1-1000) 2. Települések száma: |

* + - 1. 8. teszteset: teszt8.txt

|  |
| --- |
| Bemenet – *előfeltételnek nem megfelelő bemenet* |
| 1. 1. település 1. napi hőmérséklete: -55 |
| Kimenet |
| 1. Helytelen formátum! (-50-50)   1. település 1. napi hőmérséklete: |

* 1. Fejlesztési lehetőségek

1. A már beírt adatok módosítása utólag
2. Települések neveinek megadása