„Programozási alapismeretek”  
beadandó feladat

Készítette: *Dán Imre*Neptun-azonosító: *UUUHSK*E-mail: uuuhsk@INF.ELTE.HU

Kurzuskód: IP-18PROGEGGyakorlatvezető neve: *Daiki Tennó*

2022. 12. 18.

Tartalom

[Felhasználói dokumentáció 3](#_Toc122286536)

[Feladat 3](#_Toc122286537)

[Futási környezet 3](#_Toc122286538)

[Használat 3](#_Toc122286539)

[A program indítása 3](#_Toc122286540)

[A program használata billentyűzetről való bevitel esetén 3](#_Toc122286541)

[A program használata fájlból való bevitel esetén 3](#_Toc122286542)

[A program kimenete 4](#_Toc122286543)

[Minta bemenet és kimenet 4](#_Toc122286544)

[Hibalehetőségek 4](#_Toc122286545)

[Fejlesztői dokumentáció 5](#_Toc122286546)

[Feladat 5](#_Toc122286547)

[Tervezés 5](#_Toc122286548)

[Specifikáció 5](#_Toc122286549)

[Visszavezetés 5](#_Toc122286550)

[Algoritmus 6](#_Toc122286551)

[Fejlesztői környezet 6](#_Toc122286552)

[Forráskód 7](#_Toc122286553)

[Megoldás 7](#_Toc122286554)

[Programparaméterek 7](#_Toc122286555)

[Programfelépítés 7](#_Toc122286556)

[Függvénystruktúra 8](#_Toc122286557)

[A kód 8](#_Toc122286558)

[Tesztelés 10](#_Toc122286559)

[Érvényes tesztesetek 10](#_Toc122286560)

[Fejlesztési lehetőségek 10](#_Toc122286561)

2. Felhasználói dokumentáció
   1. Feladat

**Átlagosan legmelegebb település**

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

Készíts programot, amely megadja azt a települést, amelyen az M napra várt átlaghőmérséklet a lehető legnagyobb!

* 1. Futási környezet

IBM PC, exe futtatására alkalmas, 32-bites operációs rendszer (pl. Windows 10 Pro). Nem igényel egeret.

* 1. Használat
     1. A program indítása

A program a ConsoleApp1\ConsoleApp1\bin\Debug\netcoreapp3.1\ConsoleApp1.exe néven található a tömörített állományban.

* + 1. A program használata billentyűzetről való bevitel esetén

Az ConsoleApp1.exe fájl elindításával a program az adatokat a **billentyűzet**ről olvassa be a következő sorrendben:

|  |  |  |
| --- | --- | --- |
| # | Adat | Magyarázat |
| **1.** | *Települések száma (n)* | Természetes szám |
| **2.** | *Napok száma (m)* | Természetes szám |
| **3.** | *1. településen az 1. nap hőmérséklete* | -50 és 50 között egész szám innentől |
| **4.** | *1. településen a 2. nap hőmérséklete* |  |
| **…** | *…* |  |
|  | *1. településen az m. nap hőmérséklete* |  |
|  | *2. településen az 1. nap hőmérséklete* |  |
|  | *…* |  |
|  | *n. településen az m. nap hőmérséklete* |  |

* + 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:

ConsoleApp1.exe < be1.txt

A fájl felépítésének a következő formai követelményei vannak. A fájl első sorában a települések száma (N) és a napok száma (M) van. Az ezt követő N sorban az egyes napokra jósolt M hőmérséklet értéke található. Például:

3 5

10 15 12 10 10

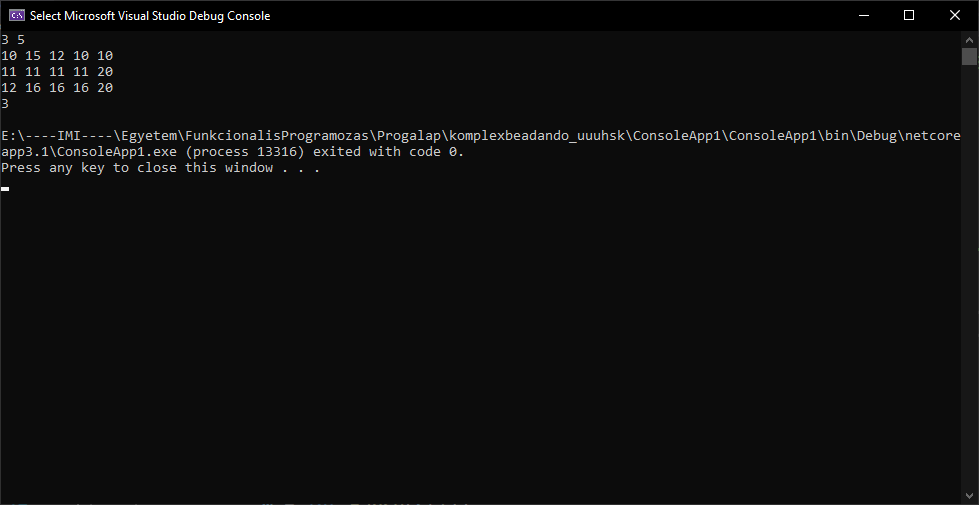
11 11 11 11 20

12 16 16 16 20

* + 1. A program kimenete

A program kiírja annak a településnek az indexét, amelyen az M napra várt átlaghőmérséklet a lehető legnagyobb.

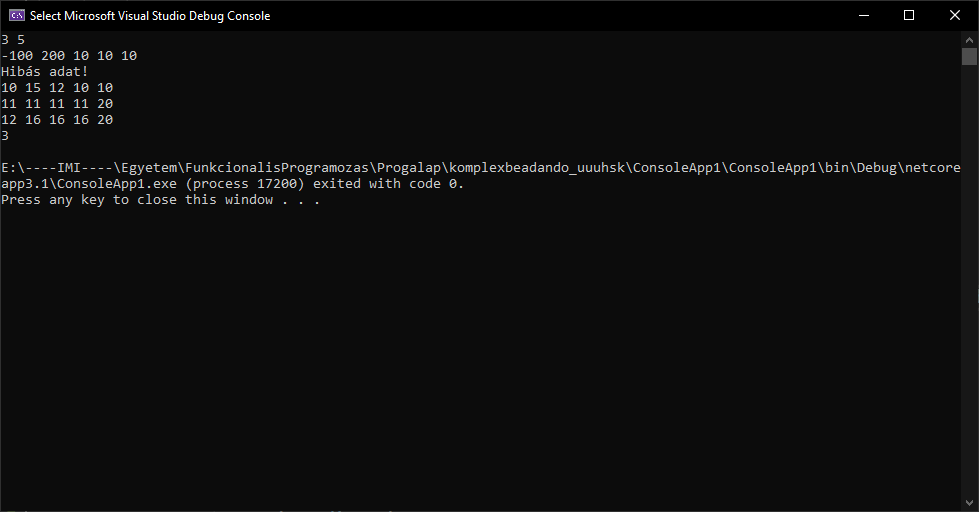
* + 1. Minta bemenet és kimenet



* + 1. Hibalehetőségek

Az egyes bemeneti adatokat a fenti mintának megfelelően kell megadni. Hiba, ha bármelyik megadandó adat nem a fenti táblázatnak megfelelő. Hiba esetén a program azzal jelzi a hibát, hogy újra kérdezi az adott sort.

Mintafutás hibás bemeneti adatok esetén:



1. Fejlesztői dokumentáció
   1. Feladat

**Átlagosan legmelegebb település**

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

Készíts programot, amely megadja azt a települést, amelyen az M napra várt átlaghőmérséklet a lehető legnagyobb!

* 1. Tervezés
  2. Specifikáció

**BE**:

**KI**:

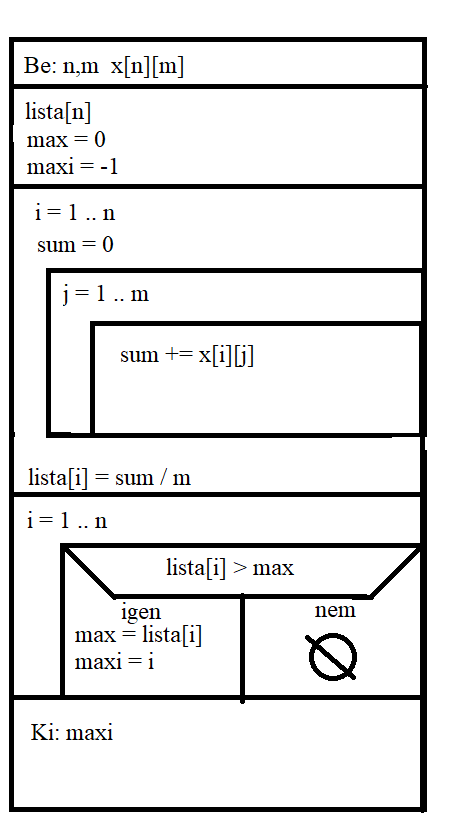
**EF**:

**UF**:

* 1. Visszavezetés

|  |  |  |
| --- | --- | --- |
| **Maximum kiválasztás** | **Előadás** | **Feladat** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Összegzés** |  |  |
|  | N | n,m |
|  | X | x |
|  | S | lista[i] |

* 1. Algoritmus



* 1. Fejlesztői környezet

IBM PC, exe futtatására alkalmas operációs rendszer (pl. Windows 10 Pro). Visual Studio 2022 (Version 17.2.3) fejlesztői környezet.

* 1. Forráskód

A teljes fejlesztői anyag –kicsomagolás után– az A1B2C3 nevű könyvtárban található meg. A fej­lesztés során használt könyvtár-struktúra:

|  |  |
| --- | --- |
| Állomány | Magyarázat |
| \ConsoleApp1\ConsoleApp1\bin\Debug\netcoreapp3.1\ConsoleApp1.exe | futtatható kód (a futtatáshoz szükséges fájlokkal) |
| \ConsoleApp1\ConsoleApp1\obj\ | mappa fordításhoz szükséges kódokkal |
|  |  |
| \ConsoleApp1\ConsoleApp1\Program.cs | C# forráskód |
|  |  |
| \ConsoleApp1\ConsoleApp1\bin\Debug\netcoreapp3.1\be1.txt | teszt-bemeneti fájl1 |
| \ConsoleApp1\ConsoleApp1\bin\Debug\netcoreapp3.1\be2.txt | teszt-bemeneti fájl2 |
|  |  |
| \Progralap komplex beadandó.docx | dokumentációk (ez a fájl) |

* 1. Megoldás
     1. Programparaméterek
        1. Konstans

1. -
   * + 1. Típus
2. -
   * + 1. Változó

x : **Tömb(**1..n**,**1..m**:Egész)**  
max : **Egész**

maxi : **Egész**

* + 1. Programfelépítés

A program által használt modulok (és helyük):

Program.cs – program, a forráskönyvtárban   
ConsoleApp1.sln – program ’megoldás fájl’, a forráskönyvtárban   
ConsoleApp1.csproj – program ’projekt fájl’, a forráskönyvtárban

* + 1. Függvénystruktúra
    2. A kód

A Program.cs fájl tartalma:

/\*

Készítette: Dán Imre

Neptun: UUUHSK

E-mail: uuuhsk@vilaghalo. uuuhsk@INF.ELTE.HU hu uuuhsk@INF.ELTE.HU

Feladat: Átlagosan legmelegebb település

\*/

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

namespace ConsoleApp1

{

class Program

{

public static string filename = "be1.txt";

public static (List<List<int>> x,int n,int m) beovasas()

{

int n = 0;

int m = 0;

List<List<int>> x = new List<List<int>>();

if (File.Exists(filename))

{

string[] s = File.ReadAllLines(filename);

foreach (var item in s)

Console.WriteLine(item);

string[] temp = s[0].Split(' ');

n = int.Parse(temp[0]);

m = int.Parse(temp[1]);

for (int i = 1; i < s.Length; i++)

x.Add(s[i].Split(' ').Select(int.Parse).ToList());

}

else

{

do

{

string[] s = Console.ReadLine().Split(' ');

n = int.Parse(s[0]);

m = int.Parse(s[1]);

if (n < 1 || m < 1 || n >= 1000 || m >= 1000)

Console.Error.WriteLine("Hibás adat!");

} while (n < 1 || m < 1 || n >= 1000 || m >= 1000);

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

{

List<int> s = new List<int>();

bool wrongData = false;

do

{

wrongData = false;

string[] temp = Console.ReadLine().Split(' ');

foreach (var item in temp)

{

int bekertszam = 100;

if (!int.TryParse(item, out bekertszam) || bekertszam < -50 || bekertszam > 50)

{

wrongData = true;

break;

}

s.Add(bekertszam);

}

if (wrongData)

Console.Error.WriteLine("Hibás adat!");

} while (wrongData);

x.Add(s);

}

}

return (x,n,m);

}

public static int feladat(List<List<int>> x, int n, int m)

{

//feladat

double max = 0;

int maxi = -1;

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

{

double tempmax = 0;

foreach (var item in x[i])

tempmax += item;

tempmax /= m;

if (tempmax <= max && maxi != -1)

continue;

max = tempmax;

maxi = i;

}

return maxi + 1;

}

static void Main(string[] args)

{

var temp = beovasas();

int solution = feladat(temp.x, temp.n, temp.m);

//Console.WriteLine("Atlagosan legmelegebb nap: " + (maxi + 1));

Console.WriteLine(solution);

}

}

}

* 1. Tesztelés
     1. Érvényes tesztesetek
        1. teszteset: be1.txt

|  |
| --- |
| Bemenet |
| 3 5  10 15 12 10 10  11 11 11 11 20  12 16 16 16 20 |
| Kimenet |
| 3 |

* + - 1. teszteset: be2.txt

|  |
| --- |
| Bemenet |
| 1000 1000  43 41 46 46 50 50 49 46 48 47 47 44 48 43 46 47 50 45 49 47 … |
| Kimenet |
| 119 |

Fejlesztési lehetőségek

1. Többszöri futtatás megszervezése
2. Települések nevének megadása
3. Grafikus visszajelzés a számolás lépéseiről