**„Programozás”  
komplex beadandó feladat**

Készítette: *Bartók Patrik Róbert*Neptun-azonosító: *MNDJ3P*E-mail: mndj3p@inf.elte.hu

Kurzuskód: IT-18PROGEGGyakorlatvezető neve: *Horváth Győző*

2025. január 12.

Tartalom

[Felhasználói dokumentáció 3](#_Toc186797335)

[Feladat 3](#_Toc186797336)

[Futási környezet 3](#_Toc186797337)

[Használat 3](#_Toc186797338)

[A program indítása 3](#_Toc186797339)

[A program használata billentyűzetről való bevitel esetén 3](#_Toc186797340)

[A program használata fájlból való bevitel esetén 3](#_Toc186797341)

[A program kimenete 4](#_Toc186797342)

[Minta bemenet és kimenet 4](#_Toc186797343)

[Hibalehetőségek 4](#_Toc186797344)

[Fejlesztői dokumentáció 5](#_Toc186797345)

[Feladat 5](#_Toc186797346)

[Tervezés 5](#_Toc186797347)

[Specifikáció 5](#_Toc186797348)

[Visszavezetés 5](#_Toc186797349)

[Algoritmus 6](#_Toc186797350)

[Fejlesztői környezet 6](#_Toc186797351)

[Forráskód 6](#_Toc186797352)

[Megoldás 7](#_Toc186797353)

[Függvénystruktúra 7](#_Toc186797354)

[A kód 7](#_Toc186797355)

[A kód (magas szintű függvényekkel) 10](#_Toc186797356)

[Tesztelés 11](#_Toc186797357)

[Automatikus tesztek (Bíró) 11](#_Toc186797358)

[Automatikus tesztek (Bíró, magas szintű függvényekkel) 12](#_Toc186797359)

[Érvényes tesztesetek 13](#_Toc186797360)

[Érvénytelen tesztesetek 13](#_Toc186797361)

[Fejlesztési lehetőségek 14](#_Toc186797362)

2. Felhasználói dokumentáció
   1. 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!

* 1. Futási környezet

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

* 1. Használat
     1. A program indítása

A program az A1B2C3\bin\Release\A1B2C3.exe néven található a tömörített állományban.

* + 1. A program használata billentyűzetről való bevitel esetén

Az A1B2C3.exe fájl elindításával a program az adatokat a **billentyűzet**ről olvassa be a következő sorrendben:

|  |  |  |
| --- | --- | --- |
| # | Adat | Magyarázat |
| **1.** | *Helységek száma (n)* | Nemnegatív szám |
| **2.** | *Madárfajok száma (m)* | Nemnegatív szám |
| **3.** | *1. helységben az 1. madárfaj száma* | Nemnegatív szám innentől |
| **4.** | *1. helységben az 2. madárfaj száma* |  |
| **…** | *…* |  |
|  | *1. helységben az m. madárfaj száma* |  |
|  | *2. helységben az 1. madárfaj száma* |  |
|  | *…* |  |
|  | *n. helységben az m. madárfaj száma* |  |

* + 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:

A1B2C3.exe < be1.txt

A fájl felépítésének a következő formai követelményei vannak. A fájl első sorában a helységek száma (n) és a madárfajok száma (m) van. A következő n sor mindegyikében m darab szám szerepel, közülük az i-edik sorban a j-edik szám az i-edik helységben a j-edik sorszámú fajból megfigyelt madarak száma. Például:

4 3

1 0 4

2 0 1

0 2 0

1 0 0

* + 1. A program kimenete

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

* + 1. Minta bemenet és kimenet

A képen szöveg látható

Automatikusan generált leírás

* + 1. 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 újra kérdezi azt.

* + - 1. Mintafutás hibás bemeneti adatok esetén:

A képen szöveg látható

Automatikusan generált leírás

1. Fejlesztői dokumentáció
   1. 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!

* 1. Tervezés
     1. [Specifikáció](https://progalap.elte.hu/specifikacio/?data=H4sIAAAAAAAACmWRTW7CMBCFrzKaVZAMSiirkVKJqj9C0HTTsglZODiAAwkVTlERYtEdvUVZsuYIuQknqewkFOhqRm%2Bev3m216jeo6EcySHP5DxFwruIID1utx6DpCxc6MZ3Go2UOY1GEgwWg7QrCURYOCbRbKXy%2FbiyidBYHpcES54mXOS7BXn12x4DrQPAn27Jmttve1bsarTZ5ksWB7d2rWIk%2BU5N5jOpyINPuOScZpZkcUGauiYpyOP3zxTyvTLM6SUzzg9XieL8oLM8d7z7f2FcG5Z8vLraZmAPI4K6bt5GBJYIWfUYNbfb6edfvZen9qsly0zn1zbRiq0MpIEhwyxSmULy1yh4xpEwJWjpWUJwYyoXBH6V2ncY2AxaATspTaM4Z4rNQItnSnGqVMxnibDkV%2FnJuJqajQxTnkRIiAwXkfqYZUjOJtj8AubX%2F8Q%2FAgAA)

Be: n∈N, m∈N, mad∈N[1..n,1..m]

Ki: db∈N, helység∈N[1..db]

Fv: vanmadár:N->L,

vanmadár(i)=**VAN**(j=1..m, mad[i,j]>0)

Fv: másholis:N x N->L,

másholis(i,j)=**VAN**(k=1..n, i≠k és mad[k,j]>0)

Fv: jó:N->L,

jó(i)=**MIND**(j=1..m, mad[i,j]=0 vagy másholis(i,j))

Ef: -

Uf: (db,helység)=**KIVÁLOGAT**(i=1..n, vanmadár(i) és jó(i), i)

* + 1. Visszavezetés

*Kiválogatás*

y ~ helység

e..u ~ 1..n

T(i) ~ vanmadár(i) és jó(i)

f(i) ~ i

*Eldöntés (vanmadár)*

i ~ j

e..u ~ 1..m

T(i) ~ mad[i,j]>0

*(Optimista) eldöntés (jó)*

i ~ j

e..u ~ 1..m

T(i) ~ mad[i,j]=0 vagy másholis(i,j)

*Eldöntés (másholis)*

i ~ k

e..u ~ 1..n

T(i) ~ i≠k és mad[k,j]>0

* + 1. [Algoritmus](https://progalap.elte.hu/stuki/?data=H4sIAAAAAAAACq1Y21LbOhT9FY6YYWCOhuMakoBpzgzXloS2QEJLw%2FBgW7IjX2TqS0Lo5GP6yDOfkB87IyXaViDMwU15ibJjrbX2VTI%2FESPIQsamsbtjbJvmjmma9R1jq95AGPWTAU1PCbJ4EUUYZTSibk5JaeEJoRmyfiKSf7guPhfnYi0BwYDRnZ1Snpeb%2BowQypHl2VFGMcpHdxRZKKM%2FCspdijBy%2BywiKeWnJEPWDRoOjNEl80yE0WGdfDnvbH1Dt2Nc2hWn9mDJqSt5hTm3cxpTniOMcnqfiz2O1TTQGKNRuH8%2BNPNr4ADDHAfoqu5dSOgP4yo2pUuAo%2Bg04CouDfssklwJJyxnCUcWYs13m5scYeQkZNSZ6Tmd82mM0f03lhf%2BQwsUgGFOAaiu7vDd1dlDRFpDhBHx28N21zmXvmc0Iezi7B6YwTDH3Oi1g36%2Bf%2FUqM0%2BSO%2BEJPKjwtJ1%2FxhMQKOQDjqLTgDU6rX4W0zHvWd4GNo9tMvmVrrONlcljthJMntbZhijWtKAHqc3d%2Flw6tYxJ6AWPQCjGuMyHEq4lSBOuoVZqI%2BL8%2FU7QQK5hQijD0jR9Go2yyaN%2FQ5xbq8kE20n%2B%2FUd61DkCNjAsP48%2B%2B27QaA8ZwuiBpZe5vZUhjEb750dGcSAarNHvXnOn15J1AU8rJdp2zW9N35v9DqymDO3RwfAy7XwKgQIMcxSa2Kou1%2FwgH7Y%2BbUmPAEfRacBVPFo0pIK1KN9rxrLOOY1XYpvcMBzcrvn5nrFodoGrY1yqVMLAMCdMi06VUAfTOoY8w4lQJn7JfA5sbjVnEZATTNURTDBl%2BBNU06FiNQc2F2S9TtrbrV%2FWgQwMy3dMN9we1GJXQG0dt6LjIBW3i%2Fb96TGvdwuEUWF0oqsLeiXr68t1wbv1hwSUgGF5JZdh63vHP%2BkijGLe36%2B1a7FALfbPgvbxAGF0cviBvzs0B1IJ6FZKNEe0BGiRqty7cdC9axmn5aACwxyFFrbKF4ydem%2FA4p70CHAUnQZcxaP%2F69111bhNY2Vg%2B6OVePIr6ycRy9YZDjY2FrUyeC5OUyUaTlNlmNOpBes3WhkKUJFoFblkemPGiehlP9%2BTrQwFrqjAsHwlTZ6spuATPFDfikcreI1H66g384Sziu05Zmv3MPLKQaEM86kp26tqxXa6zuiiNnRkxQIONIjWtxU8WlSxoaxYrk6bdfZXM5RfRPWG6thZWKzg9BiXepVEMMwnt4xTlaCH02KFGQXXtHJoLZlZee7MYiEvT2oGwuVJGZamgjEwO3fEvT9PCzdP%2FNSORfZ%2FIm7HYvcnm4kzdWCnzHYi8ZqJvk5%2BRXnyMHl6T9jg3zXuZHd7zDr2J4%2FZw%2Ft%2FhA1hlCbJi7ekrJ8Mv5ZA4uI8NX6kNqHpTPgYA7l26Z7hb1hnic9Cmy3SpOkJZs%2FjqSDtFxHmGcZLsdohrusqlT6Tr2mV7wNvUfkscq8rjVnCyetatSlVVevcMfAbmsPfi65Wqm9TfCsGRZSk8pI7rd0btOp5hmGIa%2Biq%2BPREJ696ngs2x6jBypOrhrFl2HLlUtvbaaDbaSsc2G7op0nByQzYMR3xprgqPhUwdZWNNlxbrTy5ItuuQU0BJ%2F4H8wLOdV1Xsrquq%2BA8U9moR9zX4LR2fIEq%2FmY%2Bvwnf83Z3p78axo67WxMrSuk2qesRGY%2F%2FA83xpiWFEgAA)

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás

* 1. Fejlesztői környezet

IBM PC, exe futtatására alkalmas operációs rendszer (pl. Windows 11 Home). 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 |
| A1B2C3\bin\Release\netcoreapp3.1\A1B2C3.exe | futtatható kód (a futtatáshoz szükséges fájlokkal) |
| A1B2C3\obj\ | mappa fordításhoz szükséges kódokkal |
|  |  |
| A1B2C3\Program.cs | C# forráskód |
|  |  |
| A1B2C3\teszt1.txt | teszt-bemeneti fájl1 |
| A1B2C3\teszt2.txt | teszt-bemeneti fájl2 |
| A1B2C3\teszt3.txt | teszt-bemeneti fájl3 |
| A1B2C3\teszt4.txt | teszt-bemeneti fájl4 |
| A1B2C3\teszt5.txt | teszt-bemeneti fájl5 |
|  |  |
| A1B2C3\doksi\A1B2C3.docx | dokumentációk (ez a fájl) |

* 1. Megoldás
     1. Függvénystruktúra
     2. A kód

A Program.cs fájl tartalma:

/\*

Készítette: ???

Neptun: ???

E-mail: ???

Feladat: Madármegfigyelés/ Helységek csupa máshol is előforduló madárfajjal

\*/

using System;

namespace beadando\_helysegek\_csupa\_mashol\_madarral {

internal class Program {

static void Main(string[] args) {

// deklarálás: bemenet

int[,] mad;

// deklarálás: kimenet

// statikus tömbbel dolgozunk, így szükség van a db-re is

int db;

int[] helyseg;

mad = beolvas();

(db, helyseg) = kivalogat(mad);

kiir(db, helyseg);

}

static int[,] beolvas() {

if (Console.IsInputRedirected) {

return beolvas\_biro();

}

else {

return beolvas\_kezi();

}

}

static int[,] beolvas\_biro() {

string[] sor = Console.ReadLine().Split(" ");

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

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

int[,] mad = new int[n, m];

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

sor = Console.ReadLine().Split(" ");

for (int j = 0; j < m; j++) {

mad[i, j] = int.Parse(sor[j]);

}

}

return mad;

}

static int[,] beolvas\_kezi() {

int n, m;

bool jo;

do {

Console.ResetColor();

Console.Write("Helységek száma = ");

jo = int.TryParse(Console.ReadLine(), out n) && n >= 0;

if (!jo) {

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Természetes szám kell!");

}

} while (!jo);

do {

Console.ResetColor();

Console.Write("Madárfajok száma = ");

jo = int.TryParse(Console.ReadLine(), out m) && m >= 0;

if (!jo) {

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Természetes szám kell!");

}

} while (!jo);

int[,] mad = new int[n, m];

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

for (int j = 0; j < m; j++) {

do {

Console.ResetColor();

Console.Write("{0}. helyseg {1}. madárfaj darabszáma = ", i + 1, j + 1);

jo = int.TryParse(Console.ReadLine(), out mad[i, j]) && mad[i, j] >= 0;

if (!jo) {

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Természetes szám kell!");

}

} while (!jo);

}

}

return mad;

}

static (int db, int[] helyseg) kivalogat(int[,] mad) {

int n = mad.GetLength(0);

int[] helyseg = new int[n];

int db = 0;

for (int i = 1; i <= n; i++) {

if (vanmadar(i, mad) && jo(i, mad)) {

db = db + 1;

helyseg[db - 1] = i;

}

}

return (db, helyseg);

}

static bool vanmadar(int i, int[,] mad) {

int m = mad.GetLength(1);

int j = 1;

while (j <= m && !(mad[i - 1, j - 1] > 0)) {

j = j + 1;

}

bool van = j <= m;

return van;

}

static bool jo(int i, int[,] mad) {

int m = mad.GetLength(1);

int j = 1;

while (j <= m && (mad[i - 1, j - 1] == 0 || masholis(i, j, mad))) {

j = j + 1;

}

bool mind = j > m;

return mind;

}

static bool masholis(int i, int j, int[,] mad) {

int n = mad.GetLength(0);

int k = 1;

while (k <= n && !(i != k && mad[k - 1, j - 1] > 0)) {

k = k + 1;

}

bool van = k <= n;

return van;

}

static void kiir(int db, int[] helyseg) {

if (Console.IsOutputRedirected) {

Console.WriteLine(db);

for (int i = 0; i < db; i++) {

Console.Write("{0} ", helyseg[i]);

}

Console.WriteLine();

}

else {

Console.ForegroundColor = ConsoleColor.Green;

if (db == 0) {

Console.WriteLine("Nincs a feltételnek megfelelő helység!");

}

else {

Console.WriteLine("{0} darab feltételnek megfelelő helység is van, sorszámaik:", db);

for (int i = 0; i < db - 1; i++) {

Console.Write("{0}, ", helyseg[i]);

}

Console.WriteLine(helyseg[db - 1]);

}

Console.ForegroundColor = ConsoleColor.Black;

Console.BackgroundColor = ConsoleColor.Gray;

Console.WriteLine("Kérem, nyomjon ENTER-t a folytatáshoz!");

Console.ResetColor();

Console.ReadLine();

}

}

}

}

* + 1. A kód (magas szintű függvényekkel)

A Program.cs fájl tartalma:

namespace beadando\_helysegek\_csupa\_mashol\_madarral {

internal class Program {

static void Main(string[] args) {

// deklarálás: bemenet

int[,] mad;

// deklarálás: kimenet

// statikus tömbbel dolgozunk, így szükség van a db-re is

int db;

int[] helyseg;

mad = beolvas();

int n = mad.GetLength(0);

int m = mad.GetLength(1);

helyseg = Mintak.Kivalogat(1, n, i => vanmadar(i) && jo(i), i => i);

db = helyseg.Length;

bool vanmadar(int i) {

return Mintak.Van(1, m, j => mad[i - 1, j - 1] > 0);

}

bool jo(int i) {

return Mintak.Mind(1, m, j => mad[i - 1, j - 1] == 0 || masholis(i, j));

}

bool masholis(int i, int j) {

return Mintak.Van(1, n, k => i != k && mad[k - 1, j - 1] > 0);

}

kiir(db, helyseg);

}

static int[,] beolvas() {

// ugyanaz

}

static void kiir(int db, int[] helyseg) {

// ugyanaz

}

}

}

* 1. Tesztelés
     1. Automatikus tesztek (Bíró)[[1]](#footnote-1)

A képen szöveg, képernyőkép, szám látható

Automatikusan generált leírás

* + 1. Automatikus tesztek (Bíró, magas szintű függvényekkel)

A képen szöveg, szám, képernyőkép, Betűtípus látható

Automatikusan generált leírás

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

|  |
| --- |
| Bemenet – *nincs helyseg, nincs madarfaj* |
| 0 0 |
| Kimenet |
| 0 |

* + - 1. teszteset: be2.txt

|  |
| --- |
| Bemenet – *1 helység, 1 madárfaj, 1 darab* |
| 1 1  1 |
| Kimenet |
| 0 |

* + - 1. teszteset: be3.txt

|  |
| --- |
| Bemenet – *1 helység, 1 madárfaj, nincs madár* |
| … |
|  |
| Kimenet |
| … |

* + - 1. teszteset: be4.txt

|  |
| --- |
| Bemenet – *…* |
| … |
|  |
| Kimenet |
| … |

* + - 1. teszteset: be5.txt

|  |
| --- |
| Bemenet – *…* |
| … |
| … |
| Kimenet |
| … |

* + 1. Érvénytelen tesztesetek

Billentyűzetes bevitel esetén

* + - 1. teszteset

|  |
| --- |
| Bemenet – *szöveges adat* |
| N = 11tizenegy |
| Kimenet |
| Újrakérdezés: N = |

* + - 1. teszteset

|  |
| --- |
| Bemenet – *Negatív szám* |
| N = -1 |
| Kimenet |
| Újrakérdezés:  N= |

…

* + - 1. 8. teszteset

…

* 1. Fejlesztési lehetőségek

1. Többszöri futtatás megszervezése
2. Helységek és madárfajok nevének megadása
3. Grafikus visszajelzés a számolás lépéseiről

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”. [↑](#footnote-ref-1)