„Programozás”
n. beadandó feladat

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

Kurzuskód: IKSEK-22PROGEG
Gyakorlatvezető neve: Magyar Péter

2025.12.10

1. Fázis

Időjárás előrejelzés **

Minden nap meleg települések

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 azokat a településeket, ahol minden hőmérséklet 0 fok feletti!

Bemenet

A `standard bemenet` első sorában a települések száma ($1 \le N \le 1000$) és a napok száma ($1 \le M \le 1000$) van. Az ezt követő N sorban az egyes napokra jósolt M hőmérséklet értéke található ($-50 \le H_{i,j} \le 50$).

Kimenet

A `standard kimenet` első sorába a minden nap 0 fok feletti települések T számát kell kiírni! Ezt ezen települések sorszáma kövesse, növekvő sorrendben!

Példa

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

Korlátok

Időlimit: 0.1 mp.

Memórialimit: 32 MB

a) Specifikáció

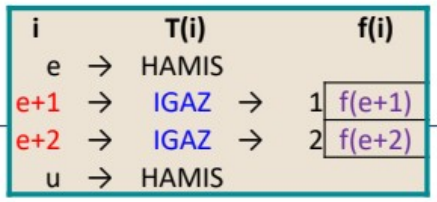
Spec

In: n ∈ N, m ∈ N, towns ∈ R[1..n*m]
Out: t ∈ N, ids ∈ R[1..t]
Pre: n >= 1 and n <= 1000 and
    m >= 1 and m <= 1000 and
    ∀i ∈ [1..n*m]:(towns[i/n + i%m] >= -50) and
    ∀i ∈ [1..n*m]:(towns[i/n + i%m] <= 50)
Post: (t,ids) = FILTER(i=1..n, ∀j ∈ [1..m]:(towns[(i-1)*m + j] > 0), i)

b) Visszavezetés

Kiválogatás

Kiválogatás sablon



Feladat

Adott az egész számok egy [e..u] intervalluma, egy ezen értelmezett T:[e..u]→Logikai feltétel és egy f:[e..u]→H függvény. Határozzuk meg az f függvény az [e..u] intervallum azon értékeinél felvett értékeit, amelyekre a T feltétel teljesül!

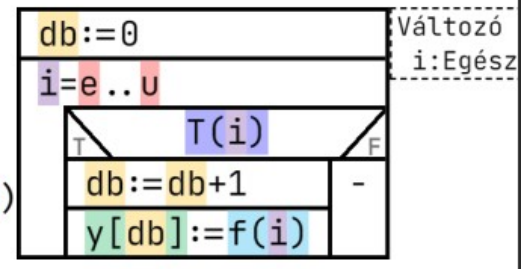
Specifikáció

```
Be: e∈Z, u∈Z
Ki: db∈N, y∈H[1..db]
Ef: -
Uf: db=DARAB(i=e..u,T(i)) és
    ∀i∈[1..db]:(
      ∃j∈[e..u]:T(j) és y[i]=f(j))
 és y⊆(f(e),f(e+1),...,f(u))
```

Rövidítve:

```
Uf: (db,y)=KIVÁLOGAT(i=e..u,T(i),f(i))
```

Algoritmus



E: 1

U: N*M

db: t

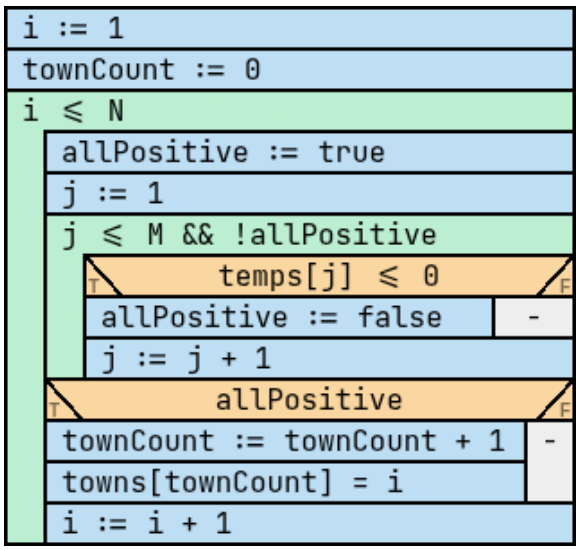
Y: ids

T(i): ∀j ∈ [1..m]:(towns[(i-1)*m + j] > 0)

f(i): i

c) Struktogram

Stuki



2. Fázis

Forráskód

```
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]} ");
                            }
                        }
                    }
                }
```