**Megoldás sablon 1. C. Beadandó Házi feladat.**

Név: Pályi Kristóf

Neptunkód: Q9NL7W

Feladat a biroból (lehet képernyőkép is)

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

1. Részteljesítés max: 50 pont

|  |  |
| --- | --- |
| Specifikáció | max: 20 pont |
| Be: n eleme N és meres eleme Meresek[1..n],  Meresek =Mennyiseg x Ar, Mennyiseg = N, Ar = N  Ki: ki eleme N  Sa: arak[1..n]  Fv: benneVan: N -> L,  benneVan(i) = NEM(VAN(j=1..n, arak[j] = i))  Ef: 1<=n<=100 és 1<=Mennyiseg,Ar<=10000  Uf: arak = MÁSOL(i=1..n, benneVan(i)) és ki = DARAB(1..n, arak[i] = arak[i]) | Itt megalkotva, vagy másolva más környezetből vagy kézzel írt megoldás fényképe |
| Sablon + forrásmegjelölés | max: 5 pont |
| <https://docs.google.com/viewerng/viewer?url=https://canvas.elte.hu/courses/38943/files/2441091/download?verifier%3DzoKuTfgFXsTxnlF5o1fszmUjjGEOr2ESCSinHDN1>  4. előadás / 10. oldal    <https://docs.google.com/viewerng/viewer?url=https://canvas.elte.hu/courses/38943/files/2423154/download?verifier%3DVxbu1M8RyDS5OAIbK3nt12PJgtufg6CzTovXFgFa>  3. előadás / 60. oldal | Lehet képernyőkép vagy másolat az előadás anyagából.  Forrásmegjelölés: a forrásfájl neve (publikálás útvonala) és oldalszám/dia sorszáma. |
| Visszavezetési táblázat (megfeleltetés) | max: 5 pont |
| Másolás:  e..u ~ 1..n  y ~ arak  f(i) ~ benneVan(i)  Megszámolás:  e..u ~ 1..n  db ~ ki  T(i) ~ arak[i] = arak[i] | A specifikáció és a minta közötti kapcsolat leírása |
| Algoritmus | max: 20 pont |
|  | Itt megalkotva vagy képernyőkép vagy kézzel írt megoldás fényképe |

**Megoldás sablon 2. C. Beadandó Házi feladat.**

FOLYTATÁSA AZ I. beadott résznek. MÁSOLJA annak a végére!

1. II. Részteljesítés max 50 pont

Módosítások max 10 pont

Itt kell dokumentálni, ha a beadott tervet a visszajelzés vagy a kódolás és tesztelés során kiderülő részletek miatt módosítani kell. A dokumentáció tartalmazza

|  |  |
| --- | --- |
| Rossz:  Be: n eleme N és meres eleme Meresek[1..n],  Meresek =Mennyiseg x Ar, Mennyiseg = N, Ar = N  Ki: ki eleme N  Sa: arak[1..n]  Fv: benneVan: N -> L,  benneVan(i) = NEM(VAN(j=1..n, arak[j] = i))  Ef: 1<=n<=100 és 1<=Mennyiseg,Ar<=10000  Uf: arak = MÁSOL(i=1..n, benneVan(i)) és ki = DARAB(1..n, arak[i] = arak[i])  Jó:  Be: evekSzam eleme N és Meres eleme Meresek[1..n],  Meresek =Mennyiseg x Ar, Mennyiseg = N, Ar = N  Ki: ki eleme N  Sa: arakRec[1..n]  Fv: Hanyszor: N -> N,  Hanyszor(elem) = DARAB(i=1..evekSzam, Meres[i].Ar = elem)  Ef: 1<=evekSzam<=100 és 1<=Mennyiseg,Ar<=10000  Uf: arakRec = MÁSOL(i=1..evekSzam, 1/Hanyszor(Meres[i].Ar)) és ki = SZUMMA(i=1..evekSzam, arakRec[i])  Sablon:    <https://docs.google.com/viewerng/viewer?url=https://canvas.elte.hu/courses/38943/files/2451951/download?verifier%3DNHFe2QsrjuPQ0KL7M0jf7YA7zliLTxqHKcDC99Og>  Programozási minták / 4. oldal  Visszavezetési táblázat:  Összegzés:  e..u ~ 1..n  f(i) ~ arakRec[i]  s ~ ki  Másolás:  e..u ~ 1..n  f(i) ~ 1/Hanyszor(Meres[i].Ar)  y ~ arakRec  Megszámolás:  e..u ~ 1..n  T(i) ~ Meres[i].Ar = elem  db ~ Hanyszor(elem)  --------------------------------------------------------------------------------------------------------  Algoritmus:  Rossz:    Jó: | Visszajelzés hatására, szinte az egész specifikációt átírtam, mivel rossz irányba indultam el.  A specifikáció módosítása miatt, a forrásokhoz és a visszavezetéshez is hozzá kell még adnom a SZUMMA függvényt, illetve a meglévőkön is változtatnom kell.  Az algoritmusom is szinte teljesen rossz, mert rossz irányba indultam el. |

(A módosítások jellegétől függően, lehet szövegdoboz nyilakkal vagy lábjegyzet is az indoklás.)

Maximális pontszámot ér az összes módosítás dokumentálása. Ha nem volt módosítás és a kód a korábban beadott tervet képezi le, akkor megkapható a maximális pontszám (nem a mennyiség, hanem a minőség számít.) Ebben az esetben is szerepeljen itt az első részteljesítés.

Kód forrása max 11 pont

|  |  |
| --- | --- |
| biro.elte.hu téma: Programozás 2. beadandó feladat: Különböző árú borok száma beadás (feltöltés) sorszáma: 2. (egyébként csak 1 van fent csak már nem tudok újat beadni pontszám: 100 (lehet nem ennyi, azért írtam ennyit, mert működik a gépemen a kód | Break, exit vagy try-catch használata esetén a 11 pontból legfeljebb 1 pont adható. |
| 1. Feltöltés: Specifikáció újraírás miatt a kódot is újra kell írjam | A hibaüzenetek és a kód függvényében, akár 80%-os eredmény is lehet maximális pontszámú |

Kód max 26 pont

2-2-2 pont: megjegyzésbe írt információk, külön választott deklarálás és kiírás.

8 pont a külön választott beolvasás/adatok eltárolása Ezen belül break, exit vagy try-catch használata esetén a 8 pontból 0 pont adható.

12 pont az algoritmus szabványos kódolása. Ezen belül break, exit vagy try-catch használata esetén a 26 pontból 0 pont adható.

using System;

using System.Collections.Generic;

class Program

{

    class Meres // prompt: class a meresek tarolasahoz

    {

        public int Mennyiseg { get; set; }

        public int Ar { get; set; }

    }

    static void Main(string[] args)

    {

        #region Adataim

        // név: Pályi Kristóf Ferenc

        // neptun-kód: Q9NL7W

        // email: q9nl7w@inf.elte.hu

        #endregion

        #region Deklaráció

        List<decimal> arakRec = new(); // prompt: lista amiben a borok arai lesznek reciprokkent

        int ki; // prompt: valtozo amiben tarolom majd a feladat kimenetet

        List<Meres> Meresek = new();

        #endregion

        #region Beolvasás

        string evek = Console.ReadLine() ?? "0";  // prompt: beolvasom, hogy hány év lesz, vagy ha nincs input legyen 0 --> nem fut le

        int evekSzam = int.Parse(evek);

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

        {

            string sor = Console.ReadLine() ?? "";

            // prompt: beolvasom a borok adatait

            string[] sorLista = sor.Split(' ');

            int Mennyiseg = int.Parse(sorLista[0]);

            int Ar = int.Parse(sorLista[1]);

            // prompt: a spacenél szétválasztom a két értéket, hogy tudjak velük dolgozni

            Meres meres = new()

            {

                Mennyiseg = Mennyiseg,

                Ar = Ar

            };

            Meresek.Add(meres);

            // prompt: a meres classt berakom a meresekbe

        }

        #endregion

        #region Feldolgozás

        for (int i = 0; i < Meresek.Count; i++)

        {

            int hanyszor = Hanyszor(Meresek[i].Ar, Meresek);

            decimal rec = 1m / hanyszor;

            arakRec.Add(rec);

            // idea: bemasolom az arakRec listaba az elemeket

            // idea: osztva annyival ahanyszor benne vannak a listaban

        }

        decimal ossz = 0m;

        for (int i = 0; i < arakRec.Count; i++)

        {

            ossz += arakRec[i];

        }

        // prompt: osszeadom az arakRec elemeit

        ki = (int)Math.Round(ossz);

        // prompt: kerekitjuk az osszeget es berakjuk a ki-be

        #endregion

        #region Kiírás

        Console.WriteLine(ki);

        #endregion

    }

    static int Hanyszor(int *elem*, List<Meres> *lista*)

    {

        int hanyszor = 0;

        for (int i = 0; i < *lista*.Count; i++)

        {

            if (*lista*[i].Ar == *elem*)

            {

                hanyszor++;

            }

        }

        // prompt: megszamolom, hogy hanyszor van benne az adott elem van a listaban

        return hanyszor;

    }

}

Teszt max 3 pont

A feladathoz adott 2 teszten túl még 3 teszt.

(A tervmódosítást alátámasztó tesztek, speciális esetek tesztjei.)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Saját 1. | Saját 2. | Saját 3 |
| Bemenet: | 3  2100 2000  200 1500  300 2000 | 2  200 1000  100 1000 | 4  500 1500  600 2000  400 1500  300 200 |
| Kimenet: | 2 | 1 | 2 |