using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace \_2024\_03\_14\_Valasztasok

{

class Program

{

static List<Kepviselo> kepviselok = new List<Kepviselo>();

static void Main(string[] args)

{

//StreamWriter g = new StreamWriter("adatok.txt");

//g.Write("egy sorba írja az adatot. ");

//g.WriteLine("A kiírt szöveg után sort fog törni.");

//g.WriteLine("új sor.");

//g.Close();

Fajlbeovlasas();

Feladat2();

Feladat3();

Feladat4();

Feladat5();

Feladat6();

Feladat7();

Console.ReadLine();

}

static void Feladat7()

{

Console.WriteLine("\n7. feladat");

StreamWriter g = new StreamWriter("kepviselok.txt");

for (int i = 1; i < 9; i++)

{

int maxIndex = MaxKeruletIndex(i);

g.WriteLine("{0}. kerület: {1} {2} - {3}",

i,kepviselok[maxIndex].vnev, kepviselok[maxIndex].knev, kepviselok[maxIndex].part);

}

g.Close();

}

static int MaxKeruletIndex(int kerulet)

{

int maxi = 0;

for(int i = 1; i<kepviselok.Count; i++)

{

if (kepviselok[i].kerulet == kerulet && kepviselok[i].szavazatSzam > kepviselok[maxi].szavazatSzam)

maxi = i;

}

return maxi;

}

static void Feladat6()

{

Console.WriteLine("\n6. feladat");

int maxSzavazat = MaximumSzavazat();

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

{

if(kepviselok[i].szavazatSzam == maxSzavazat)

Console.WriteLine("{0} {1} - {2}",kepviselok[i].vnev, kepviselok[i].knev, kepviselok[i].part);

}

}

//jegyezd meg, nézd át még a programozási tételeket

static int MaximumSzavazat()

{

int maxe = kepviselok[0].szavazatSzam;

for (int i = 1; i < kepviselok.Count; i++)

{

if (kepviselok[i].szavazatSzam > maxe)

maxe = kepviselok[i].szavazatSzam;

}

return maxe;

}

//ezt még gondold át, Vikit kérdezd meg, hogy miért van a (double)

static void Feladat5()

{

Console.WriteLine("\n5. feladat");

int szavazokSzama = SzavazokSzama();

List<string> partok = new List<string>() { "GYEP", "Gyümölcsevők Pártja", "HEP", "Húsevők Pártja","TISZ","Tejivók Szövetsége", "ZEP", "Zöldségevők Pártja", "-", "Független jelöltek" };

for (int i = 0; i < partok.Count; i+=2)

{

int partokSzavazata = PartSzavazatok(partok[i]);

double szazalek = (double) partokSzavazata / szavazokSzama \* 100;

Console.WriteLine("{0} = {1}%",partok[i+1],Math.Round(szazalek,2));

}

}

static int SzavazokSzama()

{

int osszeg = 0;

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

osszeg += kepviselok[i].szavazatSzam;

return osszeg;

}

static int PartSzavazatok(string part)

{

int osszeg = 0;

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

if(kepviselok[i].part == part)

osszeg += kepviselok[i].szavazatSzam;

return osszeg;

}

static void Feladat4()

{

Console.WriteLine("4. feladat");

int szavazokSzama = SzavazokSzama();

double szazalak = (double)szavazokSzama / 12345 \* 100;

Console.WriteLine("A választáson {0} állampolgár, a jogosultak {1}%-a vett részt.",

szavazokSzama, Math.Round(szazalak,2));

}

static void Feladat3()

{

Console.WriteLine("3. feladat");

Console.Write("Adja meg a képviselő vezetéknevét: ");

string vnev = Console.ReadLine();

Console.Write("Adja meg a képviselő keresztnevét: ");

string knev = Console.ReadLine();

int index = KepviseloIndex(vnev, knev);

if ( index == -1)

Console.WriteLine("Ilyen nevű képviselőjelölt nem szerepel a nyilvántartásban!");

else

Console.WriteLine("{0} képviselő {1} darab szavaztot kapott.", vnev + " " + knev, kepviselok[index].szavazatSzam);

Console.WriteLine();

}

static int KepviseloIndex(string vnev, string knev)

{

int i = 0;

while (i < kepviselok.Count && !(kepviselok[i].vnev == vnev && kepviselok[i].knev == knev))

i++;

if (i < kepviselok.Count)

return i;

else

return -1;

}

static void Feladat2()

{

Console.WriteLine("2. feladat");

Console.WriteLine("A helyhatósági választáson {0} képviselőjelölt indult.", kepviselok.Count);

Console.WriteLine();

}

static void Fajlbeovlasas()

{

StreamReader f = new StreamReader("szavazatok.txt");

while (!f.EndOfStream)

{

string[] st = f.ReadLine().Split(' ');

Kepviselo sv = new Kepviselo(Convert.ToInt32(st[0]), Convert.ToInt32(st[1]), st[2], st[3], st[4]);

kepviselok.Add(sv);

}

f.Close();

}

}

struct Kepviselo

{

// Matzők, jellmezők

public int kerulet, szavazatSzam;

public string vnev, knev, part;

//Konstruktor

public Kepviselo(int ujKerulet, int szavazat, string ujvnev, string ujknev, string ujpart)

{

kerulet = ujKerulet;

szavazatSzam = szavazat;

vnev = ujvnev;

knev = ujknev;

part = ujpart;

}

}

}