

## Kalapácsvetés 2016

Ebben a feladatban a 2016. évi nyári olimpiai játékokon az atlétika férfi kalapácsvetés döntőjének eredményeit kell feldolgoznia. A döntő 6 dobási sorozatból állt, de a 3. sorozat után csak az addigi legjobbak folytathatták a versenyt. A dobás távolságát centiméter pontossággal mérik. Ha a dobás érvénytelen volt, akkor az eredmény helyén az "x" karakter szerepel.

*A megoldás során vegye figyelembe a következőket:*

- ☐ *Az ékezetmentes kiírás is elfogadott!*
- ☐ *A feladat jobb megértése érdekében tanulmányozza a mintákat is!*

Készítsen konzolos programot `Kalapacsvetes2016` néven, amely az alábbi feladatokat oldja meg!

1. Hozzon létre osztályt `Versenyzo` azonosítóval!
2. Az osztály tagjaként deklaráljon egy privát **szöveges** típusú vektort, vagy listát a dobások eredményeinek tárolásához!
3. Készítse el az osztály konstruktorát, ami a forrásfájl egy sorát kapja paraméterként és a következő feladatokat látja el!
  - a. Inicializálja a `Dobások` vektort vagy listát.
  - b. Eltárolja a versenyző nevét, országát és dobásainak eredményét a megfelelő változóban/jellemzőkben.
4. Olvassa be a forrás fájl (`kalapacsvetes2016.txt`) sorait és tárolja el a `Versenyzo` típusú osztálypéldányokat (objektumokat) egy olyan adatszerkezetben, ami a további feladatok megoldását lehetővé teszi!
5. Határozza meg és írja ki a képernyőre a döntőbe jutott versenyzők számát a minta szerint!
6. A 3. dobás után csak meghatározott számú versenyző folytathatta a döntőt. Írassa ki a döntőt folytató sportolók számát! Az eredmény a minta szerint jelenítse meg a képernyőn!
7. Készítsen statisztikát a minta szerint! A döntőt folytató versenyzők érvényes, sikertelen és legjobb dobásait a minta szerint!
8. Határozza meg és írja ki a minta szerint a magyar sportoló nevét és helyezését! Feltételezheti, hogy a döntőbe csak egy magyar versenyző jutott!
9. Készítsen `kalapacsvetes2016inch.txt` néven szöveges állományt, amely csak annyiban különbözzön a forrás fájlától, hogy a dobások távolságát centiméter helyett hüvelyk (`coll`) mértékegységben, három tizedes jegyre kerekítve tartalmazza! `1coll=2,54cm`!

Minta szöveges fájl:

```
kalapacsvetes2016.txt - Jegyzetömb
Fájl Szerkesztés Formátum Nézet Súgó
Wagner Domingos;Brazília;x;71,97;72,28
Ivan Cihan;Fehéroroszország;76,13;77,43;73,48;x;77,79;76,34
Szjarhej Kalamojec;Fehéroroszország;74,22;74,17;73,7
David Söderberg;Finnország;72,3;x;74,61;74,38;x;x
Amgad Elseify Ashraf;Katar;73,88;75,4;74,45;75,2;75,46;74,25
Jevhen Vinohradov;Ukrajna;73,39;x;74,11
Wojciech Nowicki;Lengyelország;x;74,94;74,97;x;x;77,73
Pars Krisztián;Magyarország;74,77;75,15;75,28;74,89;74,62;x
Diego del Real;Mexikó;73,35;73,58;76,05;x;70,83;73,57
Serghei Marghiev;Moldova;73,31;74,14;x
Marcel Lomnický;Szlovákia;73,33;72,65;74,96;75,09;75,97;74,64
Dilsod Nazarov;Tádzsikisztán;76,16;77,27;78,07;77,17;78,68;77,68
```

Minta a programról:

5. feladat: Döntőbe jutott versenyzők száma: 12

6.feladat: A 3. dobás után 8 versenyző folytathatta a döntőt

7.feladat: Statisztika (név;érvényes dobás;sikertelen dobás;legjobb dobás

Ivan Cihan; 5; 1; 77,79 cm  
David Söderberg; 3; 3; 74,61 cm  
Amgad Elseify Ashraf; 6; 0; 75,46 cm  
Wojciech Nowicki; 3; 3; 77,73 cm  
Pars Krisztián; 5; 1; 75,28 cm  
Diego del Real; 5; 1; 76,05 cm  
Marcel Lomnický; 6; 0; 75,97 cm  
Dilsod Nazarov; 6; 0; 78,68 cm

8.feladat: A magyar versenyző Pars Krisztián 7. lett

Minta a kalapacsvetes2016inch.txt fájlról

kalapacsvetes2016inch.txt - Jegyzetkönyv

Fájl Szerkesztés Formátum Nézet Súgó

```
Wagner Domingos;Brazília;x;28,335;28,457;
Ivan Cihan;Fehéroroszország;29,972;30,484;28,929;x;30,626;30,055;
Szjarhej Kalamojec;Fehéroroszország;29,22;29,201;29,016;
David Söderberg;Finnország;28,465;x;29,374;29,283;x;x;
Amgad Elseify Ashraf;Katar;29,087;29,685;29,311;29,606;29,709;29,232;
Jevhen Vinogradov;Ukrajna;28,894;x;29,177;
Wojciech Nowicki;Lengyelország;x;29,504;29,516;x;x;30,602;
Pars Krisztián;Magyarország;29,437;29,587;29,638;29,484;29,378;x;
Diego del Real;Mexikó;28,878;28,969;29,941;x;27,886;28,965;
Serghei Marghiev;Moldova;28,862;29,189;x;
Marcel Lomnický;Szlovákia;28,87;28,602;29,512;29,563;29,909;29,386;
Dilsod Nazarov;Tádzsikisztán;29,984;30,421;30,736;30,382;30,976;30,583;
```

Megoldás:

```
using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace kalapacsvetes2016
{
    class versenyzo
    {
        public string nev { get; private set; }
        public string orszag { get; private set; }
        public List<string> dobasok { get; private set; }
        public int dobasokszama { get { return dobasok.Count - 2; } }

        public versenyzo(string sor)
        {
            string[] m = sor.Split(';');
            nev = m[0];
            orszag = m[1];
            dobasok = new List<string>();
        }
    }
}
```

```

    for (int i = 2; i < m.Length; i++)
    {
        dobasok.Add(m[i]);
    }
}

static void Main(string[] args)
{
    //Olvassa be a forrás fájl (kalapacsvetes2016.txt) sorait és tárolja el a
    //Versenyző típusú osztálpéldányokat (objektumokat)
    // egy olyan adatszerkezetben, ami a további feladatok megoldását lehetővé teszi!
    int dontobejutottakszama = 0;
    List<versenyzo> t = new List<versenyzo>();
    foreach (var i in File.ReadAllLines("kalapacsvetes2016.txt"))
    {
        t.Add(new versenyzo(i));
        Console.WriteLine(" {0} ", i); //sorok kiírása
    }
    //Határozza meg és írja ki a képernyőre a döntőbe jutott versenyzők számát a minta szerint!
    Console.WriteLine("5. feladat: Döntőbe jutott versenyzők száma: {0}", t.Count());
    /*dobások kiírása
    foreach (var i in t)
    {
        foreach (var j in i.dobasok)
        {
            Console.Write("{0} ", j);
        }
        Console.WriteLine("\n");
    }*/
    //A 3. dobás után csak meghatározott számú versenyző folytathatta a döntőt.
    //Az eredményt a minta szerint jelenítse meg a képernyőn!
    foreach (var i in t)
    {
        if (i.dobasok.Count > 3) dontobejutottakszama++;
    }
    Console.WriteLine("\n6.feladat: A 3. dobás után {0} versenyző folytathatta a döntőt",
dontobejutottakszama);
    //Készítsen statisztikát a minta szerint! A döntőt folytató versenyzők érvényes, sikertelen és
    legjobb dobásaihoz
    //készítsen a Versenyző osztályban jellemzőket az osztálydiagram szerint!
    int ervenyessedobasokszama = 0;
    int sikertelendobasokszama = 0;
    double legjobbdobas = 0;
    double[] legjobberedmenyek = new double[100];
    int sz = 0;
    string magyarsportolonev = "";

    double magyareredmeny = 0;
    Console.WriteLine("\n7.feladat: Statisztika (név;érvényes dobás;sikertelen dobás;legjobb dobás");
    foreach (var i in t)
    {
        if (i.dobasok.Count > 3)
        {
            ervenyessedobasokszama = 0;
            sikertelendobasokszama = 0;
            legjobbdobas = 0;
            foreach (var j in i.dobasok)
            {
                if (j != "x")
                {

```

```

        ervenyessedobasokszama++;
        if (double.Parse(j) > legjobbbdobas) legjobbbdobas = double.Parse(j);
    }
    else sikertelendobasokszama++;
    //Console.WriteLine("{0} ", j);
}
Console.WriteLine("\t{0}; {1}; {2}; {3} cm", i.nev, ervenyessedobasokszama,
sikertelendobasokszama, legjobbbdobas);
if (i.orszag == "Magyarország")
{
    magyarsportolonev = i.nev;
    magyareredmeny = legjobbbdobas;
}

legjobberedmenyek[sz] = legjobbbdobas;
sz++;

}
}
//Határozza meg és írja ki a minta szerint a magyar sportoló nevét és helyezését!
//Feltételezheti, hogy a döntőbe csak egy magyar versenyző jutott!
int helyezes =0;

//legjobb eredmények sorbarendezeése maximumkiválasztással
int l,r, h = sz , db = 0;
double max;
while (h > 1)
{
    l = 0; max = legjobberedmenyek[0]; r = 0;
    while (l <= h)
    {
        if (legjobberedmenyek[l] > max)
        {
            max = legjobberedmenyek[l];
            r = l;
        }
        l++;
    }
    db++;
    legjobberedmenyek[r] = legjobberedmenyek[h];
    legjobberedmenyek[h] = max;

    h--;
}
//Console.WriteLine("\nSzámok sorbarendezeve: ");
for (l = 1; l <= sz; l++)
    if (legjobberedmenyek[l] == magyareredmeny) helyezes = l;
    // Console.WriteLine("{0} ", legjobberedmenyek[l]);
Console.WriteLine("\n8.feladat: A magyar versenyző {0} {1}. lett", magyarsportolonev, sz-
helyezes+1);
//Készítsen kalapacsvetes2016inch.txt néven szöveges állományt,
//amely csak annyiban különbözzön a forrás fájlától,
//hogy a dobások távolságát centiméter helyett hüvelyk (coll) mértékegységben,
//három tizedes jegyre kerekítve tartalmazza! 1coll=2,54cm!
double eredmeny;
FileStream fnev = new FileStream("kalapacsvetes2016inch.txt", FileMode.Create);
StreamWriter fajlbairo = new StreamWriter(fnev);
foreach (var i in t)
{
    fajlbairo.Write(i.nev);
    fajlbairo.Write(";");
    fajlbairo.Write(i.orszag);

```

```

fajlbairo.Write(";");
foreach (var j in i.dobasok)
{
    if (j != "x")
    {
        eredmeny= double.Parse(j) / 2.54;
        fajlbairo.Write("{0};", Math.Round(eredmeny,3));
    }

    else fajlbairo.Write("x;");
}
fajlbairo.WriteLine("\n");
}
fajlbairo.Close();
fnev.Close();
Console.ReadKey();
}
}
}
}

```