

## Magyarország hegyei

A következő feladatban Magyarország legmagasabb hegyeinek adataival kell dolgoznia. A megoldás során vegye figyelembe a következőket:

- A képernyőre írást igénylő részfeladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például:3. feladat:)!
- Az egyes feladatokban a kiírásokat a minta szerint készítse el!
- Az ékezetmentes kiírás is elfogadott.
- A program megírásakor a fájlban lévő adatok helyes szerkezetét nem kell ellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.
- Megoldását úgy készítse el, hogy az azonos szerkezetű, de tetszőleges bemeneti adatok mellett is helyes eredményt adjon!

1. A feladat megoldásához hozzon létre grafikus vagy konzolalkalmazást (projektet) HegyekMo azonosítóval!
2. Az UTF-8 kódolású hegyekMo.txt állomány Magyarország legmagasabb hegyeinek adatait tartalmazza a következő minta szerint (forrás: wikipedia.hu):

Hegycsúcs neve;Hegység;Magasság

Ágasvár;Mátra;789

Bálvány;Bükk-vidék;956

Büszkés-hegy;Bükk-vidék;952

Cserepes-kő;Bükk-vidék;823

Az állományban a hegycsúcs nevét, a hegység megnevezését és a hegycsúcs magasságát (méter) tároltuk. Az adatokat pontosvessző választja el. Olvassa be a hegyekMo.txt állományban lévő adatokat és tárolja el egy olyan adatszerkezetben, ami a további feladatok megoldására alkalmas! A fájlban legfeljebb 1000 sor lehet! Ügyeljen arra, hogy az állomány első sora az adatok fejlécét tartalmazza!

3. Határozza meg és írja ki a képernyőre a minta szerint, hogy hány hegy található az állományban!
4. Határozza meg és írja ki a képernyőre a minta szerint az állományban található hegyek átlagmagasságát!
5. Határozza meg és írja ki a képernyőre a minta szerint a legmagasabb hegy adatait! Feltételezheti, hogy nem alakult ki holtverseny.
6. Kérjen be a felhasználótól egy magasságértéket! A bevitt adatot nem kell ellenőriznie. Döntse el, hogy található-e a megadott értéknél magasabb hegycsúcs! A keresést ne folytassa, ha a választ meg tudja adni! A képernyőre írást a minta szerint végezze!
7. Határozza meg és írja ki a képernyőre a minta szerint azoknak a hegycsúcsoknak a számát, amelyek 3000 lábnál magasabbak! Az átváltáshoz az  $1\text{ m} = 3.280839895\text{ láb}$  értékkel dolgozzon!
8. Készítsen statisztikát hegységek szerint a hegycsúcsok számáról! A megoldást úgy készítse el, hogy az inputállományba később más hegységek is bekerülhetnek! A képernyőre írást a minta szerint végezze!
9. A bukk-vidék.txt állományba írja ki azoknak a hegycsúcsoknak nevét és magasságát a minta szerint, amelyek a Bükk-vidéken magasodnak! Az állomány első sora az adatok fejlécét tartalmazza! A magasságokat egy tizedesjegyre kerekítve, lábban kell kiírnia. Az átváltáshoz az  $1\text{ m} = 3.280839895\text{ láb}$  értékkel dolgozzon! **Minta:**

3. feladat: Hegycsúcsok száma: 100 db  
 4. feladat: Hegycsúcsok átlagos magassága: 872,66 m  
 5. feladat: A legmagasabb hegycsúcs adatai:  
     Név: Kékes  
     Hegység: Mátra  
     Magasság: 1014 m  
 6. feladat: Kérek egy magasságot: 1000  
     Van 1000 m-nél magasabb hegycsúcs a Kékes  
 7. feladat: 3000 lábnál magasabb hegycsúcsok száma: 209  
 8. feladat: Hegység statisztika  
     Mátra: 25 db  
     Bükk-vidék: 60 db  
     Börzsöny: 9 db  
     Zempléni-hegység: 4 db  
     Kőszegi-hegység: 2 db  
 9. feladat: bukk-vidék.txt  
     Bálvány;3136,5  
     Büszkés-hegy;3123,4  
     Cserepes-kő;2700,1  
     Csikorgó;2552,5  
     Esztea-fő;2614,8  
     Fekete-Sár-bérc;3051,2  
     ...

### Megoldás:

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

```
namespace hegyekmo
```

```
{
    class Hegyek
    {
        public Hegyek(string sor)
        {
            string[] sorelemek = sor.Split(';');
            this.Hegycsucsnev = sorelemek[0];
            this.Hegysegnev = sorelemek[1];
            this.Magassag = Convert.ToInt32(sorelemek[2]);
        }
        public string Hegycsucsnev { get; set; }
        public string Hegysegnev { get; set; }
        public int Magassag { get; set; }
    }
    class Program
    {
        /*2. Az UTF-8 kódolású hegyekMo.txt állomány Magyarország legmagasabb hegyeinek adatait
        tartalmazza
        * a következő minta szerint (forrás: wikipedia.hu):
        Hegycsúcs neve;Hegység;Magasság
        Ágasvár;Mátra;789
        Bálvány;Bükk-vidék;956
        Büszkés-hegy;Bükk-vidék;952
        Cserepes-kő;Bükk-vidék;823
        Az állományban a hegycsúcs nevét, a hegység megnevezését és a hegycsúcs magasságát (méter) tároltuk.
        Az adatokat pontosvessző választja el. Olvassa be a hegyekMo.txt állományban lévő adatokat és
        tárolja el egy olyan adatszerkezetben,
        ami a további feladatok megoldására alkalmas! A fájlban legfeljebb 1000 sor lehet!
        Ügyeljen arra, hogy az állomány első sora az adatok fejlécét tartalmazza! */
        public static List<Hegyek> hegycsucsadatok = new List<Hegyek>();
        static void Main(string[] args)
        {
            StreamReader olvas = new StreamReader("hegyekMo.csv", Encoding.UTF8);
            string fejléc = olvas.ReadLine();//ha van fejléc
            while (!olvas.EndOfStream)
            {
                hegycsucsadatok.Add(new Hegyek(olvas.ReadLine()));
            }
        }
    }
}
```

```

}
int Hegyekszama = hegycsucsadatok.Count;
int i;//ciklusváltozó

/*Console.WriteLine("Az adatok listája fájlból");
Console.WriteLine("Hegycsúcs neve           Hegység           Magasság");
for (i = 0; i < Hegyekszama; i++)
{
    Console.WriteLine("{0,-25} {1,-20} {2}", hegycsucsadatok[i].Hegycsucsnev,
hegycsucsadatok[i].Hegysegnev, hegycsucsadatok[i].Magassag);
}*/
//3. Határozza meg és írja ki a képernyőre a minta szerint, hogy hány hegy található az
állományban!
Console.WriteLine("3. feladat: Hegycsúcsok száma: {0} db", Hegyekszama);
//4.Határozza meg és írja ki a képernyőre a minta szerint az állományban található
hegyek átlagmagasságát!
//összegzés tétele
double atlagmagassag = 0;
for (i = 0; i < Hegyekszama; i++)
{
    atlagmagassag += hegycsucsadatok[i].Magassag;
}
Console.WriteLine("4. feladat: Hegycsúcsok átlagos magassága: {0} m",
atlagmagassag/Hegyekszama);

//5. Határozza meg és írja ki a képernyőre a minta szerint a legmagasabb hegy adatait!
//Feltételezheti, hogy nem alakult ki holtverseny.
//maximumkiválasztás tétele
int max = hegycsucsadatok[0].Magassag;
int maxi = 0;
for (i = 0; i < Hegyekszama; i++)
{
    if (hegycsucsadatok[i].Magassag > max)
    {
        max = hegycsucsadatok[i].Magassag;
        maxi = i;
    }
}

Console.WriteLine("5. feladat: A legmagasabb hegycsúcs adatai:\n\tNév: {0}\n\tHegység:
{1}\n\tMagasság: {2} m",
    hegycsucsadatok[maxi].Hegycsucsnev, hegycsucsadatok[maxi].Hegysegnev,
hegycsucsadatok[maxi].Magassag);
/*6. Kérjen be a felhasználótól egy magasságértéket!
* A bevitt adatot nem kell ellenőriznie.
* Döntse el, hogy a található-e a megadott értéknél magasabb hegycsúcs!
* A keresést ne folytassa, ha a választ meg tudja adni!
* A képernyőre írást a minta szerint végezze!*/
//keresés tétele
int keresettmagassag;
Console.Write("6. feladat: Kérek egy magasságot: ");
keresettmagassag = Convert.ToInt32(Console.ReadLine());
bool van;
i = 0;
while ((i<Hegyekszama) && !(hegycsucsadatok[i].Magassag > keresettmagassag))
{
    i++;
}
van = i < Hegyekszama ? true : false;
if (van)
{
    Console.WriteLine("\tVan {0} m-nél magasabb hegycsúcs a
{1}", keresettmagassag, hegycsucsadatok[i].Hegycsucsnev);
}

else
{
    Console.WriteLine("\tNincs {0} m-nél magasabb hegycsúcs", keresettmagassag);
}

```

```

//7.Határozza meg és írja ki a képernyőre a minta szerint azoknak a hegycsúcsoknak a
számát,
//amelyek 3000 lábnál magasabbak!Az átváltáshoz az 1 m = 3.280839895 láb értékkel
dolgozzon!
//megszámolás tétele
int db = 0;
for (i = 0; i < Hegyekszama; i++)
{
    if (hegycsucsadatok[i].Magassag* 3.280839895 > 3000)
    {
        db++;
    }
}
Console.WriteLine("7. feladat: 3000 lábnál magasabb hegycsúcsok száma: {0}", db);
/*8. Készítsen statisztikát hegységek szerint a hegycsúcsok számáról!
* A megoldást úgy készítse el, hogy az inputállományba később más hegységek is
bekerülhetnek!
* A képernyőre írást a minta szerint végezze! */
Console.WriteLine("8. feladat: Hegység statisztika");
hegycsucsadatok.GroupBy(x => x.Hegysegnev).ToList().ForEach(x =>
Console.WriteLine("\t{0}: {1} db",x.Key,x.Count()));

/*9. A bukk-vidék.txt állományba írja ki azoknak a hegycsúcsoknak nevét és magasságát a
minta szerint,
* amelyek a Bükk-vidéken magasodnak! Az állomány első sora az adatok fejlécét
tartalmazza!
* A magasságokat egy tizedesjegyre kerekítve, lábban kell kiírnia.
* Az átváltáshoz az 1 m = 3.280839895 láb értékkel dolgozzon! */
Console.WriteLine("9. feladat: bukk-vidék.txt");
FileStream fnev = new FileStream("bukk-vidék.txt", FileMode.Create);
StreamWriter fajlbairo = new StreamWriter(fnev);
double magassaglab;
fajlbairo.WriteLine("Hegycsúcs neve;Magasság láb");
for (i = 0; i < Hegyekszama; i++)
{
    if(hegycsucsadatok[i].Hegysegnev== "Bükk-vidék")
    {
        magassaglab =Math.Round((hegycsucsadatok[i].Magassag * 3.280839895),1);
        fajlbairo.Write("{0};", hegycsucsadatok[i].Hegycsucsnev);
        fajlbairo.Write("{0}", magassaglab);
        Console.Write("\t{0};", hegycsucsadatok[i].Hegycsucsnev);
        Console.WriteLine("{0}", magassaglab);
        fajlbairo.WriteLine("\n");//sortörés
    }
}
fajlbairo.Close();
fnev.Close();

Console.ReadKey();

}
}
}

```