

## Nobel-díjak

A következő feladatban a Svéd Királyi Tudományos Akadémia által osztott Nobel-díj napjainkig feljegyzett adatait tartalmazó szöveges állományt kell feldolgoznia. 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!

I. A feladat megoldásához hozzon létre grafikus vagy konzolalkalmazást (projektet) Nobel azonosítóval!

2. Az UTF-8 kódolású nobel . csv állomány tartalmazza a kiosztott díjak adatait. Minden adatsorhoz rendre a következő mezők tartoznak: • évszám
  - típus (fizikai, kémiai, orvosi, irodalmi, béke, közgazdaságtani) • keresztnév
  - vezetéknév

Az adatokat pontosvessző választja el egymástól. Ügyeljen arra, hogy a fájl első sora az adatok fejlécét tartalmazza! Amennyiben a díjat egy szervezet kapta, akkor a keresztnév mezőben szerepel a szervezet teljes neve, míg a vezetéknév mező ebben az esetben üres. Olvassa be a nobel . csv állományban található adatokat és tárolja el egy megfelelően megválasztott adatszerkezetben!

3. Határozza meg és Írja ki a képernyőre a minta szerint, hogy Arthur B. McDonald milyen típusú díjat kapott! Feltételezheti, hogy életében csak egyszer kapott Nobeldíjat.
4. Határozza meg és írja ki a képernyőre a minta szerint, hogy ki kapott 2017-ben irodalmi Nobel-díjat!
5. Határozza meg és írja ki a képernyőre a minta szerint, hogy mely szervezetek kaptak béke Nobel-díjat 1990-től napjainkig!
6. A Curie család több tagja is kapott díjat. Határozza meg és írja ki a képernyőre a minta szerint, hogy melyik évben a család melyik tagja milyen díjat kapott!
7. Határozza meg és írja ki a képernyőre a minta szerint, hogy melyik típusú díjból hány darabot osztottak ki a Nobel-díj történelme folyamán!
8. Hozzon létre orvosi . txt néven egy UTF-8 kódolású szöveges állományt, amely tartalmazza az összes kiosztott orvosi Nobel-díj adatait! A fájlban a minta szerint szerepeljen kettősponttal elválasztva a kiosztás éve és a díjazott teljes neve! A sorok évszám szerint növekvő rendben legyenek az állományban!

Minta:

2. feladat:

Összesen 923 Nóbeldíjas adatait tartalmazza a nobel.csv

3. feladat: fizikai

4. feladat: Ishiguro Kazuo

5. feladat:

2017 International Campaign to Abolish Nuclear Weapons (ICAN)

2015 National Dialogue Quartet

2013 Organisation for the Prohibition of Chemical Weapons (OPCW)

2012 European Union (EU)

2007 Intergovernmental Panel on Climate Change (IPCC)

2006 Grameen Bank

2005 International Atomic Energy Agency (IAEA)

2001 United Nations (U.N.)

1999 Médecins Sans Frontières

1997 International Campaign to Ban Landmines (ICBL)

1995 Pugwash Conferences on Science and World Affairs

6. feladat:

1935 :Joliot-Curie Irène (kémiai)

1911 :Curie, née Skłodowska Marie (kémiai)

1903 :Curie Pierre (fizikai)

1903 :Curie, née Skłodowska Marie (fizikai)

7. feladat:

orvosi 214 db

fizikai 207 db

kémiai 178 db

béke 131 db

irodalmi 114 db

közgazdaságtani 79 db

8. feladat: orvosi.txt

## Példa az *orvosi.txt* egy lehetséges részletére:

```
1990:Joseph E. Murray
1990:E. Donnall Thomas
1991:Erwin Neher
1991:Bert Sakmann
1992:Edmond H. Fischer
1992:Edwin G. Krebs
1993:Richard J. Roberts
1993:Phillip A. Sharp
1994:Alfred G. Gilman
1994:Martin Rodbell
1995:Edward B. Lewis
1995:Christiane Nüsslein-Volhard
1995:Eric F. Wieschaus
1996:Peter C. Doherty
1996:Rolf M. Zinkernagel
1997:Stanley B. Prusiner
1998:Robert F. Furchgott
1998:Louis J. Ignarro
1998:Ferid Murad
1999:Gunter Blobel
```

Megoldás:

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

namespace nobel
{
    class Program
    {
        struct adatsor//évszám;tipus;vezetéknév;keresztnev
        {
            public int evszam;
            public string tipus;
            public string vnev;
            public string knev;
        }

        static adatsor[] adatok = new adatsor[1000];//maximum 1000 adatot tartalmazhat az adatok
        static void Main(string[] args)
        {
            string[] fajlbol = File.ReadAllLines("nobel.csv");//adatok beolvasása fájlból
            int i,j;
            int tindex = 0;//sorok száma a fájlban
            //adatok tárolása a struktúrában
            for (i = 1; i < fajlbol.Count(); i++)//a ciklus 1-től indul, mert az első sor a
mezőneveket tartalmazza
            {
                string[] egysordarabolva = fajlbol[i].Split(';');
                adatok[tindex].evszam = Convert.ToInt32(egysordarabolva[0]);
                adatok[tindex].tipus = egysordarabolva[1];
                adatok[tindex].knev = egysordarabolva[2];
                adatok[tindex].vnev = egysordarabolva[3];
                tindex++;
            }
            /*
            //adatok kiírása a képernyőre
            Console.WriteLine("évszám      típus      név");
            for (i = 0; i < tindex; i++)
                Console.WriteLine("{0,-5}{1,-20}{2} {3}", adatok[i].evszam, adatok[i].tipus,
adatok[i].vnev, adatok[i].knev);
            */
            //2. feladat
            //Határozza meg és írja ki a képernyőre,
            //hogyan hány díj adatot tartalmazza a nobel.csv!
            Console.WriteLine("2. feladat:\nÖsszesen {0} Nóbeldíjas adatot tartalmazza a
nobel.csv", tindex);
            //3. feladat Arthur B.      McDonald milyen típusú díjat kapott
            //keresés tétele
            i = 0;
            while (i < tindex && String.Compare(adatok[i].knev, "Arthur B.") != 0 &&
String.Compare(adatok[i].vnev, "McDonald") != 0)
            {
                i++;
            }
            Console.WriteLine("\n3. feladat:");
            if (i < tindex)
                Console.WriteLine(" {0}", adatok[i].tipus);
            else
                Console.WriteLine("Nincs Arthur B. McDonald név.");

            //4. feladat ki kapott 2017-ben irodalmi Nóbeldíjat
        }
    }
}
```

```

i = 0;
while (i < tindex && String.Compare(adatok[i].tipus, "irodalmi") != 0)
{
    i++;
}
Console.WriteLine("\n4. feladat:");
if (i < tindex)
    Console.WriteLine(" {0} {1}", adatok[i].vnev, adatok[i].knev);
else
    Console.WriteLine("Nincs ilyen adat");

//5. feladat mely szervezetek kaptak béke nóbél díjat napjainkig
Console.WriteLine("\n5. feladat:");
for (i = 0; i < tindex; i++)
{
    if(adatok[i].evszam>=1990 && String.Compare(adatok[i].tipus, "béke") == 0 &&
adatok[i].vnev.Length==0)
    {
        Console.WriteLine("{0,-5}{1} {2}", adatok[i].evszam, adatok[i].vnev,
adatok[i].knev);
    }
}

//6. feladat Curie családból ki mikor milyen díjat kapott
Console.WriteLine("\n6. feladat:");
for (i = 0; i < tindex; i++)
{
    if ( adatok[i].vnev.ToLower().Contains("curie"))
    {
        Console.WriteLine("{0,-5}:{1} {2} ({3})", adatok[i].evszam, adatok[i].vnev,
adatok[i].knev, adatok[i].tipus);
    }
}

//7. feladat melyik típusú díjból hányat osztottak ki
Console.WriteLine("\n7. feladat:");
int kulonbozoelemekszama = 0;
string[] dijak = new string[1000];
int[] dijakszama = new int[1000];
for (i = 0; i < tindex; i++)
{
    j = 0;
    while ((j <= kulonbozoelemekszama) && (adatok[i].tipus != dijak[j]))
    {
        j++;
    }
    if (j > kulonbozoelemekszama)
    {
        dijak[kulonbozoelemekszama] = adatok[i].tipus;
        kulonbozoelemekszama++;
    }
}
//megszámolás
for (i = 0; i < tindex; i++)
    for (j = 0; j < kulonbozoelemekszama; j++)
        if (String.Compare(adatok[i].tipus, dijak[j]) == 0)
            dijakszama[j]++;

for (i = 0; i < kulonbozoelemekszama; i++)
    Console.WriteLine("{0,-20}{1,-5} ", dijak[i], dijakszama[i]);
/*
        //sorbarendezés díjak száma szerint
        //sorbarendezés min kiválasztással

```

```

        int mini = 0;
        int mindijakszama;
        for (i = 0; i < kulonbozoelemekszama; i++)
        {
            mini = i;
            mindijakszama = dijakszama[i];
            for (j = i; j < kulonbozoelemekszama; j++)
            {
                if (dijakszama[j] > mindijakszama)
                {
                    mini = j;
                    mindijakszama = dijakszama[j];
                }
            }
            //csere
            var s = dijak[i];
            dijak[i] = dijak[mini];
            dijak[mini] = s;
            int db = dijakszama[i];
            dijakszama[i] = dijakszama[mini];
            dijakszama[mini] = db;
        }
        Console.WriteLine("díjak száma szerint sorbarendevezve");
        for (i = 0; i < kulonbozoelemekszama; i++)
            Console.WriteLine("{0,-20}{1,-5} ", dijak[i], dijakszama[i]);
        */
//8. feladat hozzon létre orvosi.txt néven egy szöveges állományt
//amely tartalmazza az összes kiosztott orvosi Nóbeldíj adatait
//a sorok évszám szerint növekvő rendben legyenek
//fájlbaírás
FileStream fnev = new FileStream("orvosi.txt", FileMode.Create);
StreamWriter fajlbairo = new StreamWriter(fnev);
for (i = tindex; i >=0; i--)
{
    if (String.Compare(adatok[i].tipus, "orvosi") == 0)
    {
        fajlbairo.Write("{0}:{1} {2}", adatok[i].evszam, adatok[i].vnev,
adatok[i].knev);
        fajlbairo.WriteLine("\n");//sortörés
    }
}
fajlbairo.Close();
fnev.Close();

Console.ReadKey();
}
}
}

```