

A kémiai elemek kémiailag tovább már nem bontható, egyszerű anyagok. Jelenleg 118 különböző kémiai elemet ismerünk, közülük a Földön 94 található meg a természetben, ezeket természetes kémiai elemeknek hívjuk. Ebben a feladatban a kémiai elemek felfedezésével kapcsolatos adatforrással kell dolgoznia².

A feladat megoldása során vegye figyelembe a következőket:

- *A képernyőre írást igénylő részfeladatok eredményének megjelenítése előtt írja a képernyőre a feladat sorszámát (például:5. feladat:)!*
- *Az egyes feladatokban a kiírásokat a minta szerint készítse el!*
- *Az ékezetmentes azonosítók és kiírások is elfogadottak.*
- *Az azonosítókat kis- és nagybetűkkel is kezdheti.*
- *A program megírásakor az állományban 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!*

A felfedezések.txt UTF-8 kódolású forrásállomány soraiban egy-egy felfedezés adatait tároltuk a következő sorrendben:

- Év, felfedezés ideje, például: Ókor, vagy a felfedezés évszáma
- Név, például: Ezüst
- Vegyjel, például: Ag
- Rendszám, például: 47
- Felfedező(k), például: W.Ramsay és J.W.S.Rayleigh

Az állomány első sora a mezőneveket tartalmazza, az adatokat pontosvesszővel választottuk el:

```
Év;Név;Vegyjel;Rendszám;Felfedező
Ókor;Arany;Au;79;Ismeretlen
Ókor;Ezüst;Ag;47;Ismeretlen
...
1250;Arzén;As;33;Albertus Magnus
1450;Antimon;Sb;51;B. Valentinius (W. Tölde)
1520;Cink;Zn;30;Paracelsus
...
```

1. Készítsen grafikus vagy konzolalkalmazást (projektet) a következő feladatok megoldásához, amelynek projektjét Felfedezések néven mentse el!
2. Hozzon létre saját osztályt Elem azonosítóval, melynek adattagjait felhasználva egy-egy felfedezés (elem) adatait tudja majd tárolni! A feladatot megoldhatja saját osztály definiálása nélkül is, de megoldása ebben az esetben nem teljes értékű.

3. Készítse el a saját osztálykonstruktorát, mely hívásával az osztály adattagjait tudja inicializálni! Ügyeljen arra, hogy a felfedezés ideje nem csak évszám lehet!
4. Olvassa be a `felfedezések.txt` állomány sorait és tárolja az adatokat egy `Elem` osztálytípusú összetett adatszerkezetben! Ügyeljen arra, hogy az állomány első sora az adatok fejlécét tartalmazza!
5. Határozza meg és írja ki, hogy hány elem adatai találhatóak a forrásállományban!
6. Számolja meg, hogy hány elem felfedezésében működött közre William Ramsay (W.Ramsay)! Az eredményt jelenítse meg!
7. Kérjen be egy vegyjelet, majd a bemeneti adatot tárolja szöveges típusú változóban!
8. Határozza meg és írja ki, hogy a bekért vegyjelhez tartozó elem mikor került felfedezésre! Oldja meg, hogy a keresés ne legyen érzékeny a kis- és nagybetűkre! Ha a bekért adat nem található meg a vegyjelek között, akkor a „Nincs ilyen elem az adatforrásban!” szöveg jelenjen meg!
9. Készítsen statisztikát a minta szerint a 19. század éveiben felfedezett elemek számáról! A 19. század 1801-ben kezdődött és 1900-ig tartott. Az évszámok sorrendje tetszőleges!
10. Hozzon létre `felfedezések_rendszam.txt` néven UTF-8 kódolású szöveges állomány! Az állomány szerkezete és tartalma egyezzen meg a forrásállományéval azzal a különbséggel, hogy a felfedezések a **rendszám** szerint növekvő rendben legyenek a kiadott minta szerint!

Képernyő minta:

```

5. feladat: Elemek száma: 117
6. feladat: William Ramsay: 4 db
7. feladat: Kérek egy vegyjelet: pb
8. feladat: Keresés
           Felfedezés: Ókor
9. feladat: Statisztika 19. század
           1808: 4 db
           1879: 4 db
           1898: 5 db

```

Minta `felfedezések_rendszam.txt`:

```

Év;Név;Vegyjel;Rendszám;Felfedező
1766;Hidrogén;H;1;Henry Cavendish
1868;Hélium;He;2;P.J.C.Jansen 1869 J.N.Locker
1817;Lítium;Li;3;J. A. Arvedson
1797;Berillium;Be;4;L. N. Vaquelin
1808;Bór;B;5;J. L. Gay-Lusac és Thénard
Ókor;Szén;C;6;Ismeretlen
...
2010;Tennesszin;Ts;117;Egyesített Atomkutató Intézet, Dubna
2006;Oganesson;Og;118;Egyesített Atomkutató Intézet, Dubna ...

```

Megoldás:

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

namespace Felfedezesek
{
    class elemek
    {
        /*Év;Név;Vegyjel;Rendszám;Felfedező
        Ókor;Arany;Au;79;Ismeretlen*/
        public int evszam { get; set; }
        public string evszamstring { get; set; }
        public string nev { get; set; }
        public string vegyjel { get; set; }
        public int rendszam { get; set; }
        public string felfedezo { get; set; }
        public elemek(string sor)
        {
            string[] sorelemek = sor.Split(';');
            evszamstring = sorelemek[0];
            if (sorelemek[0]=="Ókor") evszamstring = sorelemek[0];
            else evszam = Convert.ToInt32(sorelemek[0]);

            nev = sorelemek[1];
            vegyjel = sorelemek[2];
            rendszam = Convert.ToInt32(sorelemek[3]);
            felfedezo = sorelemek[4];
        }
    }
}

class Program
{
    public static List<elemek> adatok = new List<elemek>();
    static void Main(string[] args)
    {
        StreamReader olvas = new StreamReader("felfedezesek.txt", Encoding.UTF8);
        string fejléc = olvas.ReadLine();//ha van fejléc
        while (!olvas.EndOfStream)//ciklus amíg nincs vége a fájlnek
        {
            adatok.Add(new elemek(olvas.ReadLine()));
        }
        int adatokszama = adatok.Count();
        for (int i = 0; i < adatokszama; i++)
        {
            Console.WriteLine("{0,-5}{1,-5}{2,-10}{3,-5}{4,-5}{5}", adatok[i].evszamstring,
            adatok[i].evszam, adatok[i].nev, adatok[i].vegyjel, adatok[i].rendszam, adatok[i].felfedezo);
        }
        Console.WriteLine("5. feladat: Elemek száma: {0}", adatokszama);
        int db = 0;
        for (int i = 0; i < adatokszama; i++)
        {
            if (adatok[i].felfedezo.Contains("W.Ramsay"))
            {
                db++;
            }
        }
        Console.WriteLine("6. feladat: William Ramsay: {0}", db);
        Console.WriteLine("7. feladat: Kérek egy vegyjelet: ");
        string keresettvegyjel = Console.ReadLine();
        int j = 0;
        while(j<adatokszama && !adatok[j].vegyjel.ToLower().Contains(keresettvegyjel.ToLower()))
        {
            j++;
        }
    }
}
```

```

        if(j<adatokszama)
        {
            Console.WriteLine("8. feladat: Keresés\n\tFelfedezés: {0}",
adatok[j].evszamstring);
        }
        else
        {
            Console.WriteLine("8. feladat: Keresés\n\tNincs ilyen elem az adatforrásban");
        }

        Console.WriteLine("9. feladat: Statisztika 19. század");
        List<elemek> ujadatok = new List<elemek>();
        for (int i = 0; i < adatokszama; i++)
        {
            if (adatok[i].evszamstring!="Ókor")
            {
                ujadatok.Add(adatok[i]);
            }
        }
        ujadatok.Where(x => x.evszam >= 1801 && x.evszam <= 1900).GroupBy(x =>
x.evszam).Where(x=>x.Count(>3)).ToList().ForEach((x => Console.WriteLine("\t{0}: {1}", x.Key,
x.Count())));
        List<elemek> sorba = adatok.OrderBy(x => x.rendszam).ToList();
        FileStream fajl = new FileStream("felfedezesek_rendszam.txt", FileMode.Create,
FileAccess.Write, FileShare.None);
        StreamWriter sw = new StreamWriter(fajl, Encoding.UTF8);
        sw.WriteLine("Év;Név;Vegyjel;Rendszám;Felfedező");
        for (int i = 0; i < adatokszama; i++)
        {
            sw.WriteLine("{0};{1};{2};{3};{4}", sorba[i].evszamstring, sorba[i].nev,
sorba[i].vegyjel, sorba[i].rendszam, sorba[i].felfedezo);
        }
        sw.Close();
        fajl.Close();
        Console.ReadKey();
    }
}
}

```