

16.1

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

namespace _2017_03_08_A_D_K
{
    class Program
    {
        static void Main(string[] args)
        {
            List<double> szamok = new List<double>();
            Console.Write("Hany_szamrol_lesz_szo:");
            int n = int.Parse(Console.ReadLine());
            for (int i = 0; i < n; i++)
            {
                Console.Write("Kerem_a_{0}_szamot:", i +
1);

                string s = Console.ReadLine();
                double d = double.Parse(s.Replace('.',
', '));

                szamok.Add(d);
            }
            //
            foreach (double x in szamok)
            Console.Write("{0}_", x);
            Console.WriteLine();
            //
            double sum = 0;
            foreach (double x in szamok)
                sum = sum + x;
            Console.WriteLine("Atlag={0}", sum / n);
        }
    }
}
```

C:\WINDOWS\system32\cmd.exe

```
Hany_szamrol_lesz_szo:5
Kerem_a_1_szamot:1
Kerem_a_2_szamot:2
Kerem_a_3_szamot:3
Kerem_a_4_szamot:4
Kerem_a_5_szamot:5
1_2_3_4_5_
Atlag=3
Press any key to continue . . .
```

Segítség a megoldáshoz: A feladat egy egyszerű ciklussal megoldható, lényegében problémamentes. Ha valaki jobb minőségű kódot szeretne készíteni, legfeljebb arra kell ügyelnie, hogy a beírt szám valóban tört alakú legyen. Tudni kell, hogy magyar területi beállítások mellett a *Double.parse* a törtszámokban a tizedesvessző megjelenésére számít, és nem a tizedespontra. A bekérésben esetleg előfordulhat tizedespont vesszőre tudjuk cserélni a *Replace* metódus hívásával.

16.2

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

```
namespace _2017._03._08_A_D_K_2
{
    class Program
    {
```

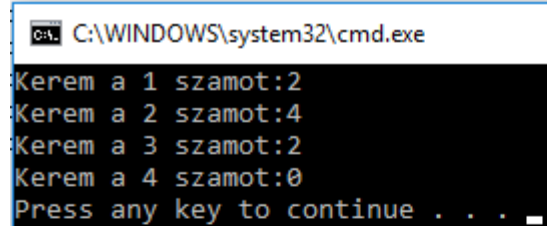
```

static void Main(string[] args)
{
    List<double> szamok = new List<double>();
    int i = 0;
    while (true)
    {
        Console.WriteLine("Kerem a {0} szamot:", i +
1);

        string s = Console.ReadLine();
        double d = double.Parse(s.Replace('.',
', '));

        if (d == 0.0) break;
        else szamok.Add(d);
        i++;
    }
}

```



```

C:\WINDOWS\system32\cmd.exe
Kerem a 1 szamot:2
Kerem a 2 szamot:4
Kerem a 3 szamot:2
Kerem a 4 szamot:0
Press any key to continue . . .

```

Segítség a megoldáshoz : A vége jel kezelése egy középen tesztelős ciklussal a leglátványosabb, legegyszerűbb. Ismert tény, hogy sokan ódzkodnak a középen tesztelős ciklustól, a *break* használatától. Megpróbálhatjuk megírni ezt a ciklust is akár elől tesztelős, akár hátul tesztelős tisztán logikai ciklussal, amelyben nincs *break*, de azt fogjuk találni, hogy a kilépési tesztet $x==0.0$ kétszer is be kell írunk. Ekkor pedig kitörhet a lázadás a kódredundancia oldaláról. Nem törünk lándzsát egyik megoldás mellett sem, és ellene sem foglalunk állást.

16.3

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

namespace _2017_03_08_A_D_K_3
{
    class Program
    {
        static void Main(string[] args)
        {
            List<double> szamok = new List<double>();
            int i = 0;
            while (true)
            {
                Console.WriteLine("Kerem a {0} szamot:", i +
1);

                string s = Console.ReadLine();
                if (s == "*" || s == "vege") break;
                double d = double.Parse(s.Replace('.', '
', ' '));

                szamok.Add(d);
                i++;
            }
        }
    }
}
```

```
C:\WINDOWS\system32\cmd.exe
Kerem a 1 szamot:2
Kerem a 2 szamot:43
Kerem a 3 szamot:4
Kerem a 4 szamot:3
Kerem a 5 szamot:*
Press any key to continue . . .
```

16.4 Feladat

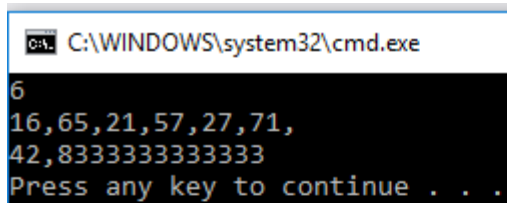
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace _2017._03._22._1F
{
    class Program
    {
        static void Main(string[] args)
        {
            int n = int.Parse(Console.ReadLine());
            List<double> szamok = new List<double>();
            Random rnd = new Random();
            for (int i = 0; i < n; i++)
```

```

        {
            if (i % 2 == 0) szamok.Add(rnd.Next(10,
51));
            else szamok.Add(rnd.Next(40, 81));
        }
        foreach (double x in szamok)
            Console.Write("{0},", x);
        double sum = 0;
        foreach (double x in szamok)
            sum = sum + x;
        Console.WriteLine();
        Console.WriteLine("{0}", sum / n);
    }
}

```



```

C:\WINDOWS\system32\cmd.exe
6
16,65,21,57,27,71,
42,83333333333333
Press any key to continue . . .

```

Feladat:[Feltöltés véletlen számokkal] egy listát töltünk fel véletlen egész számokkal oly módon, hogy a páros sorszámú listaelemek a [10...50], a páratlan sorszámú listaelemek a [40...80]intervallumból kerüljenek ki! A listában kerüljön be n ilyen szám, az n értékét a program indulásakor kérje be! A program a végén írja vissza az adatokat a képernyőre, a számokat egymás mellé írva, vesszővel elválasztva, majd adja meg a számok átlagát!

16.5 Feladat

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

namespace _2017._03._22._2F
{
    class Program
    {
        static void Main(string[] args)
        {
            int sum = 0;
            foreach (int x in szamok)
                sum = sum + x;
            Console.WriteLine("Atlag = {0}", (double)sum /
n);
        }
    }
}
```

Feladat: [Feltöltés véletlen növekvő számokban] Egy listát töltünk fel véletlen egész számokkal oly módon, hogy az első listaelem a [10...30] intervallumban essen, a következő listaelemek az őket megelőző elemtől legyenek „valamennyivel” nagyobbak, a különbség az [1...5] intervallumból kerüljön ki! A lista hosszát (n darab) a program induláskor kérje be! A program a végén írja vissza az adatokat a képernyőre, a számokat egymás mellé írva, vesszővel elválasztva, majd adja meg a számok átlagát!

16. Feladat

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

namespace _2017._0._3._22._3F
{
    class Program
    {
        static void Main(string[] args)
        {
            List<int> szamok = new List<int>();
            Random rnd = new Random();
            int x = rnd.Next(10, 31);
            for(int i= 0;1< new;i++)
            {
                szamok.Add(x);
                x = x + rnd.Next (1,6);
            }
        }
    }
}
```

16.7Feladat

Egy text fájl soronként egy törtszámot tartalmaz. Amikor több szám is szerepel egy sorban, akkor vesszővel vannak elválasztva. A vessző után szóközők is szerepelhetnek a jobb vizuális tördelés érdekében. Írjuk olyan programot, amely bekéri a fájl nevét, majd beolvassa a számokat a fájlból! A bevitel véget ér, ha a fájl minden sorát beolvastuk, vagy üres sort olvastunk be a fájlból. A program a bevitel végén írja vissza az adatokat a képernyőre, a számokat egymás mellé írva, vesszővel elválasztva, majd adja meg a számok átlagát.

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



```

namespace feladat7
{
    class Program
    {
        static void Main(string[] args)
        {
            List<double> szamok = new List<double>();

            int szamlalo = 0;

            string fajlNev = Console.ReadLine();

            StreamReader f = new StreamReader(fajlNev,
Encoding.Default);
            while (f.EndOfStream == false)
            {
                string s = f.ReadLine();
                if (s.Trim().Length == 0) break;
                double d = double.Parse(s.Replace('.',
','));

                szamok.Add(d);
            }
            f.Close();

            foreach (double x in szamok)
            {
                Console.WriteLine(x);
                szamlalo = szamlalo + 1;
            }

            Console.WriteLine();

            double sum = 0;
            foreach (double x in szamok)
                sum = sum + x;
            Console.WriteLine("{0}", sum / szamlalo);
        }
    }
}

```

```
C:\WINDOWS\system32\cmd.exe
SZAMOK.TXT
2,3
1,4
1,85
Press any key to continue . . .
```

16.8 Feladat

Egy egész számokat tartalmazó listát töltünk fel billentyűzetről oly módon, hogy a program kezelője egy időben egyszerre több számot is beírhat, vesszővel elválasztva! Ekkor minden számot adjunk hozzá a listához! Ha csak egy számot írna be a kezelő, akkor azt az egy számot. A bevitelt akkor fejezzük be, ha a kezelő egyetlen számot se ír be (üres string)! A program a bevitel végén írja vissza az adatokat a képernyőre, a számokat egymás mellé írva, vesszővel elválasztva, majd adja meg a számok átlagát!