

# **P**ROGRAMOZÁS **A**LAPFELADATOK

2011 - 2012

## TARTALOMJEGYZÉK

Beolvasás, kiírás, eldöntés, számítások.....	3
Egyszerű ciklusok.....	4
Egymásba ágyazott ciklusok:.....	6
Tömbök kezelése.....	8
Szövegkezelés.....	9
File-kezelés (I. rész):.....	10
File-kezelés (II. rész):.....	11
Keresés, kiválogatás.....	12
Megoldások (programkódok).....	13

A példatár a programozásban használatos alap adattípusok (egész és valós szám, karakter, szöveg és az ezekből álló tömbök), programozási elemek (szekvencia, elágazás, ciklus), és egyszerű programozási tételek (sorozatszámítás, maximumkeresés, keresés, kiválogatás, rendezés) megértéséhez tartalmaz feladatokat.

A feladatok sorszámozása folyamatos. Egyes fejezetek elején példakódok vagy példa algoritmusok találhatóak, ezek szürke háttérrel jelennek meg. Hasonlóan egyes feladatok esetén példa képernyőképek is láthatóak. Ha a feladat előtt az **(m)** jelölés látható, akkor ennek a feladatnak a megoldása a megoldások részben C# kódként megtalálható.

Egyes feladatokhoz forrásállományok is tartoznak, ezek a feladatgyűjteményhez tömörített állományban (`forrasok.zip`) mellékletként csatolva lettek.

Jó programozást!

Surányi Szabolcs

## Beolvasás, kiírás, eldöntés, számítások

1. Írj programot, ami kiírja a képernyőre, hogy "Hello world!"!
2. Írj programot, beolvassa a felhasználó nevét, majd köszön neki!
3. Írj programot, ami beolvas egy számot, majd kiírja a kétszeresét!
4. Írj programot, ami beolvas két számot, majd kiírja:
  - a. az összegüket;
  - b. különbségüket;
  - c. szorzatukat;
  - d. hányadosukat, ha lehet!
5. Írj programot, mely beolvas két egész számot, és kiírja a képernyőre a nagyobbikat!
6. Írj programot, mely beolvas három egész számot, és kiírja a képernyőre a legkisebbet!
7. Írj programot, ami beolvassa a háromszög oldalainak hosszát, és megmondja, hogy ilyen oldalakkal szerkeszthető-e háromszög!
8. Írj programot, mely beolvas két pozitív egész számot, és kiírja a számtani és mértani közepüket! A gyökvonáshoz használd a `Math.Sqrt()` függvényt!
9. Írj programot, mely beolvassa a másodfokú egyenlet együtthatóit, és kiírja, hogy az egyenletnek van-e megoldása!
10. Írj programot, ami beolvassa a másodfokú egyenlet együtthatóit, és kiírja az egyenlet megoldásait, ha vannak!
11. Írj programot, mely beolvassa egy derékszögű háromszög két befogóját, és megadja az átfogójának a hosszát! Az átfogót 2 tizedesjeggyel add meg!
12. Írj programot, mely beolvassa a téglalast három élének hosszát, és kiírja a felszínének és térfogatának mérőszámát!
13. Írj programot, mely beolvassa egy kör átmérőjét, és kiírja a kör területét és területét! A pi értékének meghatározásához használd a `Math.PI` értéket!
14. Írj programot, ami beolvassa a körív sugarát és középponti szögét, és kiírja a körív területét és a határoló ív hosszát!

## Egyszerű ciklusok

15. Írj programot, mely beolvas egy pozitív egész számot, és kiírja az egész számokat a képernyőre eddig a számig, egymástól szóközzel elválasztva!
16. Írj programot, mely beolvas egy pozitív egész számot, és kiírja az egész számokat egymás alá a képernyőre eddig a számig!
17. Írj programot, mely beolvas egy pozitív egész számot, és kiírja az osztóit!
18. Írj programot, mely beolvas egy pozitív egész számot, és kiírja az osztóinak az összegét!
19. Írj programot, mely beolvas egy pozitív egész számot, és megmondja, hogy tökéletes szám-e! (A tökéletes számok azok, melyek osztóinak összege egyenlő a szám kétszeresével. Ilyen szám pl. a 6, mert  $2 \cdot 6 = 1 + 2 + 3 + 6$ .)
20. Írj programot, mely beolvassa a hatvány alapját és a kitevőt, és kiírja a hatványértéket!

Példa:

Hatványalap: 2  
Kitevő: 3  
Hatványérték: 8

21. (m) Írj programot, ami csak pozitív számot hajlandó beolvasni.
22. Írj programot, mely addig olvas be számokat a billentyűzetről, ameddig azok kisebbek, mint tíz. Írja ki ezek után a beolvasott számok összegét!
23. Írj programot, amely beolvas egy egész számot, majd elosztja 2-vel annyiszor, ahányszor lehet és közben felírja a számot a kettes számok szorzataként megszorozva egy olyan számmal, amely már nem osztható 2-vel. Például:

Kérek egy egész számot: 120  
 $120 = 2 \cdot 2 \cdot 2 \cdot 15$

Ha a szám egyszer sem osztható kettővel, akkor:

Kérek egy egész számot: 17  
 $17 = 17$

24. (m) Írj programot, ami csak az "alma" szót hajlandó beolvasni, ha ez sikerült, akkor kiírja, hogy az "Az alma gyümölcs!".
25. Írj programot, ami beolvas egy egész számot, majd addig von ki belőle 3-at, amíg háromnál kisebb nem lesz az eredmény. Írja ki ezek után a hárommal való maradékos osztását a számnak. Például:

Kérek egy egész számot: 16  
 $16 = 5 \cdot 3 + 1$

26. (m) Írj programot, mely eldönti egy számról, hogy prímszám-e.
27. (m) Írj programot, mely beolvas egy számot, és kiírja a prímszámokat eddig a számig!
28. Írj programot, mely beolvas egy számot, és kiírja a szám prímosztóit!
29. Írj programot, ami beolvas egy számot, majd kiírja a szám prímtényező felbontását!
30. Írj programot, ami beolvas két számot, és kiírja a két szám legnagyobb közös osztóját! Segítség:

```

1. algoritmus: A kisebbik számtól visszafelé nézzük, hogy van-e közös
osztó. Nincs, ha az egyet elértük.

2. algoritmus:
Eljárás LNKO
  Ciklus amíg b > 0
    t := b
    b := a mod b
    a := t
  Ciklus vége
  visszaad a
Eljárás vége
    
```

31. Írj programot, ami beolvas két számot, és kiírja a legkisebb közös többszörösüket!

**Egymásba ágyazott ciklusok:**

32. Írj programot, ami beolvas egy számot, és kiírja az ilyen alapú szorzótáblát!
33. Írj programot, ami beolvas egy számot, és kiírja az ilyen alapú összegztáblát!
34. Párosával speciális szám a 12 és 42, mivel  $12 \cdot 42 = 504 = 21 \cdot 24$ , azaz a két szám szorzata nem változik, ha számjegyeiket felcseréljük. Vajon van-e még ilyen kétjegyű számpár, és melyek azok? A feladat a keresett párokat előállító program írása! Egy szám első számjegyét megkapod, ha elosztod maradékosan 10-zel ( $x/10$ ), a második számjegyét, ha veszed a tízes maradékát ( $x\%10$ ), ha az  $x$  egy kétjegyű, egész típusú szám.
35. Írj programot, ami kiírja a kisbetűket, és melléjük az ASCII kódjukat! A kiírás több oszlopos legyen, és legfeljebb 10 soros. Minta:

a 97	f 102	.....
b 98	g 103	
c 99	h 104	
d 100	i 105	
e 101	j 106	

36. Írj programot, ami a következő sormintával tölti ki a képernyő egy részét:

XOXOXO
OXOXOX
XOXOXO
OXOXOX
XOXOXO
OXOXOX

A sorok és az oszlopok számát is olvassa be a program! ügyelj arra, hogy a mintában a páros sorok o-val, a páratlanok x-el kezdődnek!

37. Kérjünk be egy természetes számot ( $a$ ), majd rajzoljunk ki a képernyőre egy háromszöget csillagokból (\*). A háromszög  $a$  sornyi csillagból álljon.  
Például  $a=4$ -re:

*
***
*****
*****

38. (m) Kérjünk be egy természetes számot (a), majd rajzoljunk ki a képernyőre egy háromszöget csillagokból (\*). A háromszög a sornyi csillagból álljon.  
Például a=4-re:

```

      *
     ***
    *****
   *********
  
```

39. Kérjünk be két természetes számot (M,N), majd rajzoljunk ki a képernyőre egy MxN méretű téglalapot csillag (\*) jelekből úgy, hogy a téglalap belseje üres legyen.  
Például M=8 és N=4-re:

```

*****
*       *
*       *
*       *
*****
  
```

40. Írj programot, mely bekér egy egész számot, és kiírja eddig a számig a tökéletes számokat!
41. Írj programot, mely kiírja a Vigenère-táblát! Az első sor az angol ABC betűit tartalmazza, majd minden további sorban az ABC az előző sorhoz képet egyel eltolva szerepel.

```

ABCDEFGHIJKLMNOPQRSTUVWXYZ
BCDEFGHIJKLMNOPQRSTUVWXYZA
CDEFGHIJKLMNOPQRSTUVWXYZAB
DEFGHIJKLMNOPQRSTUVWXYZABC
EFGHIJKLMNOPQRSTUVWXYZABCD
FGHIJKLMNOPQRSTUVWXYZABCDE
.....
  
```

## Tömbök kezelése

42. Olvass be pár számot (a darabszámot is kérje be a program), majd írd ki a páratlanok darabszámát.
43. Olvass be egy pár számot (a darabszámot is kérje be a program), majd írd ki a párosok összegét.
44. Olvass be egy pár számot (a darabszámot is kérje be a program), majd írd ki a párosokat a képernyőre, a sorszámukkal együtt, vagyis hogy hányadiknak olvastad be őket.
45. Olvassunk be egész számokat egy tömbbe (a darabszámot is kérje be a program), majd kérjünk be egy egész számot. Keressük meg a tömbben az első ilyen egész számot, majd írjuk ki a tömbindexét. Ha a tömbben nincs ilyen szám, írjuk ki, hogy a beolvasott szám nincs a tömbben.
46. Olvassunk be egész számokat egy tömbbe (a darabszámot is kérje be a program), majd kérjünk be egy egész számot, és mondjuk meg, hogy hányszor szerepel a tömbben.
47. Olvassuk be egy tömbbe az osztály tanulóinak keresztnéveit (a darabszámot is kérje be a program). Mondjuk meg, hogy egy adott keresztnevű tanulóból hány jár az osztályba (a keresett nevet is kérje be a program).
48. Olvassunk be egész számokat egy tömbbe (a darabszámot is kérje be a program), majd adjuk meg a legkisebb és a legnagyobb szám különbségét!
49. Olvassunk be egész számokat egy tömbbe (a darabszámot is kérje be a program), majd írjuk ki a számokat nagyság szerint növekvő sorrendben.  
A buborékos rendezés algoritmus (Az X tömb elemeit rendezi növekvő sorrendbe:

```
Ciklus I = N-től 2-ig
  Ciklus J = 1-től I - 1-ig
    Ha X[J] > X[J + 1] akkor
      Csere(X[J], X[J + 1])
    Elágazás vége
  Ciklus vége
Ciklus vége
```



## Szövegkezelés

50. Olvass be egy szót, és írd ki a betűit egy sorba, egymástól szóközzel elválasztva.
51. Tabu: olvass be egy szót és a tabu betűt, amit nem szabad kiírni. Írd ki a szót a tabu betű nélkül.
52. Olvass be egy szót, és írd ki a minden második betűjét egymás alá!
53. Olvass be egy szót, és írd ki a képernyőre egymás alá a karaktereinek ASCII kódját!
54. Olvass be egy szót, és írd ki a betűit fordított sorrendben!
55. Olvass be egy mondatot, és írd ki a szavait egymás alá. (A szavakat egymástól a szóköz karakter választja el.)
56. Olvass be egy szót, és írd ki a betűit fordított sorrendben, csupa nagy betűvel!
57. Olvass be egy mondatot, és írd ki a szavait egymás alá úgy, hogy a szavak első betűje nagybetűs legyen!

## File-kezelés (I. rész):

File-ból olvasás példa (az első sor tartalmazza a beolvasandó adatok számát):

```
using System;
using System.IO;
class Program
{
    static public void Main()
    {
        FileStream fs = new FileStream("forras.be", FileMode.Open);
        StreamReader rs = new StreamReader(fs);
        string s = rs.ReadLine(); //Az első sor tartalmazza a darabszámot
        int db = Convert.ToInt32(s);
        int[] szamok = new int[db];
        for (int i = 0; i < db; i++)
        {
            s = rs.ReadLine();
            szamok[i] = Convert.ToInt32(s);
        }
        rs.Close();
        fs.Close();
        Console.ReadKey();
    }
}
```

58. (m) Olvasd be a `forras58.be` állományban található adatokat (ezek egész számok, az első sor a darabszámot tartalmazza), és írd ki a képernyőre a beolvasott számok közül a legnagyobbat!
59. (m) Olvasd be a `forras59.be` állományban található adatokat (ezek egész számok, az első sor a darabszámot tartalmazza), és írd ki a képernyőre a beolvasott számok közül a legkisebb párosat!
60. Olvasd be a `forras60.be` állományban található adatokat (ezek egész számok, az első sor a darabszámot tartalmazza), és írd ki a képernyőre a beolvasott számok közül a páratlanok átlagát!
61. (m) Olvasd be a `forras61.be` állományban található adatokat (ezek szavak, az első sor a darabszámot tartalmazza), és írd ki a képernyőre a beolvasott szavak közül a leghosszabbat!
62. Olvasd be a `forras62.be` állományban található adatokat (ezek szavak, az első sor a darabszámot tartalmazza), és írd ki a képernyőre a beolvasott azon szavakat, melyek "a" betűvel kezdődnek!
63. Olvasd be az `forras63.be` állományban található egész számokat (az első sor tartalmazza a darabszámot), és írd ki ezeket a képernyőre csökkenő sorrendben!
64. (m) Olvasd be az `forras64.be` állományban található szavakat (az első sor tartalmazza a darabszámot), és írd ki őket a képernyőre ABC sorrendben!

## File-kezelés (II. rész):

File-ból olvasás példa (nem tudjuk a beolvasandó adatok számát):

```
using System;
using System.IO;
class Program
{
    static public void Main()
    {
        FileStream fs = new FileStream("forras.be", FileMode.Open);
        StreamReader rs = new StreamReader(fs);
        int[] szamok = new int[100]; //legfeljebb 100 számot olvashatunk be
        while (!r.EndOfStream) //addig olvas, míg a file végére nem ér
        {
            s = rs.ReadLine();
            szamok[i] = Convert.ToInt32(s);
        }
        rs.Close();
        fs.Close();
        Console.ReadKey();
    }
}
```

65. Olvasd be a `forras65.be` állományban található számokat, és írd ki a képernyőre a beolvasott számok közül a legkisebbet!
66. Olvasd be a `forras66.be` állományban található számokat, és írd ki a képernyőre a beolvasott számok közül a legnagyobb párosat!
67. Olvasd be a `forras67.be` állományban található számokat, és írd ki a képernyőre a beolvasott számok közül a párosak összegét!
68. Olvasd be a `forras68.be` állományban található szavakat, és írd ki a képernyőre a beolvasott szavak közül a legrövidebbet!
69. Olvasd be a `forras69.be` állományban található szavakat, és írd ki a képernyőre a beolvasott azon szavakat, melyek nem "a" betűvel kezdődnek!

## Keresés, kiválogatás

70. Az `forras70.be` állományban legfeljebb 300 pozitív egész szám található, minden sorban pontosan egy szám.
- Olvasd be az `forras70.be` állományban található számokat, és add meg az összegüket és az átlagukat! Az összeget 100-ra kerekítve add meg!
  - Add meg, hogy hány prímszám található az állományban!
  - Add meg a legkisebb prímszám előtt található páros számok átlagát!
  - Add meg az első és a második prímszám között található számok összegét!
  - Add meg a legkisebb és a legnagyobb szám sorszámát!
  - Add meg a számok terjedelmét! (A legnagyobb és a legkisebb különbségét.)
  - Add meg a legnagyobb és a legkisebb szám közötti számok közül a hárommal oszthatóak átlagát!
  - Add meg a legkisebb páros és legnagyobb páratlan szám közötti számok közül a páratlanok átlagát!
  - Add meg annak a számnak a sorszámát, amelyik a legközelebb van a 900-hoz!
  - Add meg a 900-hoz legközelebb eső szám utáni számok között a prímszámok összegét!
  - Add meg a második legnagyobb szám prímosztóit!
  - Add meg a legkisebb hárommal osztható és a legnagyobb öttel osztható szám közötti számok közül a páratlanok legnagyobb prímosztóinak az összegét!
  - Add meg az egymást követő számok egymástól való eltéréseinek az átlagát!
  - Add meg a legnagyobb olyan szám osztóinak a számát, aminek van 5-nél kisebb prímosztója!
  - Add meg a legtöbb prímosztóval rendelkező szám osztóinak összegét!

## Megoldások (programkódok)

Írj programot, ami beolvas két számot, majd kiírja:  
az összegüket;  
különbségüket;  
szorzatukat! (Saját függvények megadása)

```
using System;
namespace szamol
{
    class szamol
    {
        static int osszead(int a, int b)
        {
            return a + b;
        }
        static int kivon(int a, int b)
        {
            return a - b;
        }
        static int szor(int a, int b)
        {
            return a * b;
        }
        static void Main()
        {
            Console.Write("Első szám: ");
            string line = Console.ReadLine();
            int a = Convert.ToInt32(line);
            Console.Write("A második szám: ");
            line = Console.ReadLine();
            int b = Convert.ToInt32(line);
            int ossz = osszead(a, b);
            Console.Clear();
            Console.WriteLine("Az összeg: " + ossz);
            Console.ReadKey();
        }
    }
}
```

Írj programot, ami csak pozitív számot hajlandó beolvasni. (Hátultesztelős ciklus)

```
using System;
namespace csakPozLehet
{
    class csakPozLehet
    {
        static void Main()
        {
            string strLine;
            int intA;
            do
            {
                Console.Clear();
                Console.Write("A szám: ");
                strLine = Console.ReadLine();
                intA = Convert.ToInt32(strLine);
                if (intA <= 0)
                {
                    Console.WriteLine("Pozitív számot írj be!");
                }
            } while (intA > 0);
        }
    }
}
```

Írj programot, ami csak az "alma" szót hajlandó beolvasni, ha ez sikerült, akkor kiírja, hogy az "Az alma gyümölcs!". (Hátultesztelős ciklus)

```
using System;
namespace alma
{
    class alma
    {
        static void Main()
        {
            string line = "";
            do
            {
                Console.Clear();
                Console.Write("Írd be, hogy alma : ");
                line = Console.ReadLine();
            } while (line != "alma");
            Console.WriteLine("Az alma gyümölcs");
            Console.ReadKey();
        }
    }
}
```

Írj programot, mely eldönti egy számról, hogy prímszám-e.

```
using System;
namespace joprím
{
    class joprím
    {
        static bool prim_e(int a)
        {
            int i = 2;
            double gyok = Math.Round(Math.Sqrt(a));
            while (a%i != 0 && i <= gyok)
            {
                i++;
            }
            if (i > gyok)
            {
                return true;
            }
            else
            {
                return false;
            }
        }
        static void Main()
        {
            Console.WriteLine("A szám: ");
            string line = Console.ReadLine();
            int a = Convert.ToInt32(line);
            if (prim_e(a))
            {
                Console.WriteLine(a + " prímszám.");
            }
            else
            {
                Console.WriteLine(a + " nem prímszám.");
            }
            Console.ReadKey();
        }
    }
}
```



Írj programot, mely beolvas egy számot, és kiírja a prímszámokat eddig a számig!

```
using System;
namespace primek
{
    class primek
    {
        static bool prim_e(int a)
        {
            int i = 2;
            double gyok = Math.Round(Math.Sqrt(a));
            while (a%i != 0 && i <= gyok)
            {
                i++;
            }
            if (i > gyok)
            {
                return true;
            }
            else
            {
                return false;
            }
        }
        static void Main()
        {
            Console.Write("A szám: ");
            string line = Console.ReadLine();
            int a = Convert.ToInt32(line);
            for (int i = 2; i <= a; i++)
            {
                if (prim_e(i))
                {
                    Console.WriteLine(i);
                }
            }
            Console.ReadKey();
        }
    }
}
```

Kérjünk be egy természetes számot (a), majd rajzoljunk ki a képernyőre egy háromszöget csillagokból (\*). A háromszög a sornyi csillagból álljon.

```
using System;
namespace piramis
{
    class piramis
    {
        static void Main()
        {
            Console.Write("a: ");
            int a = Convert.ToInt32(Console.ReadLine());
            for (int i = 1; i <= a; i++)
            {
                for (int j = 1; j <= a - i; j++)
                {
                    Console.Write(" ");
                }
                for (int j = 1; j <= 2 * i - 1; j++)
                {
                    Console.Write("*");
                }
                Console.WriteLine();
            }
            Console.ReadKey();
        }
    }
}
```

**Szövegkezelés példák:**

```
using System;
namespace szoveg_darabol
{
    class szoveg_darabol
    {
        static void Main()
        {
            string s = "valamilyen valami";
            int poz_v = s.IndexOf("v");//első v helye, 0-tól számozódik
            Console.WriteLine("Az első v betű helye:" + poz_v); //0

            int poz_x = s.IndexOf("x");//első x helye, 0-tól számozódik
            Console.WriteLine("Az első x betű helye:" + poz_x);// -1, nincs a szövegben

            int poz_utolso_a = s.LastIndexOf("a");//utolsó "a" helye, 0-tól számozódik
            Console.WriteLine("Az utolsó a betű helye:" + poz_utolso_a);

            Console.WriteLine(s.Substring(0,3));//első 3 karakter, 0.-tól 3 db

            Console.WriteLine(s.Substring(11));//12. karaktertől

            int poz = s.IndexOf(" ");
            Console.WriteLine(s.Substring(0,poz));
            Console.WriteLine(s.Substring(poz + 1));

            Console.ReadKey();
        }
    }
}
```

Olvasd be az r61.be állományban található szavakat (az első sor tartalmazza a darabszámot), és írd ki őket a képernyőre ABC sorrendben!

```
using System;
using System.IO;
namespace r70
{
    class r70
    {
        static void Main()
        {
            FileStream f = new FileStream("forras70.be", FileMode.Open);
            StreamReader r = new StreamReader(f);
            int db = Convert.ToInt32(r.ReadLine());
            string[] szavak = new string[db];
            for (int i = 0; i < db; i++)
            {
                szavak[i] = r.ReadLine();
            }
            r.Close();
            f.Close();
            Console.WriteLine("A tömb:");
            for (int i = 0; i < db; i++)
            {
                Console.WriteLine(szavak[i]);
            }
            Console.ReadKey();
            Console.Clear();
            string seged;
            for (int i = db - 1; i > 0; i--)
            {
                for (int j = 0; j < i ; j++)
                {
                    if (String.Compare(szavak[j],szavak[j + 1]) > 0)
                    {
                        seged = szavak[j];
                        szavak[j] = szavak[j + 1];
                        szavak[j + 1] = seged;
                    }
                }
            }
            Console.Clear();
            Console.WriteLine("A tömb rendezve:");
            for (int i = 0; i < db; i++)
            {
                Console.WriteLine(szavak[i]);
            }
            Console.ReadKey();
        }
    }
}
```

Olvasd be a `forras58.be` állományban található adatokat (ezek egész számok, az első sor a darabszámot tartalmazza), és írd ki a képernyőre a beolvasott számok közül a legnagyobbat! (A számokat egy tömbben tároljuk.)

```
using System;
using System.IO;
namespace szamok2
{
    class Program
    {
        static void Main(string[] args)
        {
            FileStream f = new FileStream("forras1.be", FileMode.Open);
            StreamReader r = new StreamReader(f);
            int db = Convert.ToInt32(r.ReadLine());
            int[] szamok = new int[db];
            szamok[0] = Convert.ToInt32(r.ReadLine());
            int max = szamok[0];
            for (int i = 1; i < db; i++)
            {
                szamok[i] = Convert.ToInt32(r.ReadLine());
                if (szamok[i] > max)
                {
                    max = szamok[i];
                }
            }
            r.Close();
            f.Close();
            Console.WriteLine("A legnagyobb beolvasott szám: " + max);
            Console.ReadKey();
        }
    }
}
```

Olvasd be a `forras59.be` állományban található adatokat (ezek egész számok, az első sor a darabszámot tartalmazza), és írd ki a képernyőre a beolvasott számok közül a legkisebb párosat! (A beolvasott számokat nem tárolja el a program.)

```
using System;
using System.IO;
namespace szamok2
{
    class Program
    {
        static void Main(string[] args)
        {
            FileStream f = new FileStream("forras59.be", FileMode.Open);
            StreamReader r = new StreamReader(f);
            int db = Convert.ToInt32(r.ReadLine());
            int szam;
            int min = int.MaxValue; //kezdeti érték legyen a lehető legnagyobb
            int pdb = 0; //Párosok darabszáma
            for (int i = 0; i < db; i++)
            {
                szam = Convert.ToInt32(r.ReadLine());
                if (szam % 2 == 0) //csak akkor növeljük az értéket, ha páros
                {
                    pdb++;
                    if (szam < min) //kisebb, mint min
                    {
                        min = szam;
                    }
                }
            }
            r.Close();
            f.Close();
            if (pdb > 0)
            {
                Console.WriteLine("A legkisebb páros beolvasott szám: " + min);
            }
            else
            {
                Console.WriteLine("Nincs páros a beolvasottak között.");
            }
            Console.ReadKey();
        }
    }
}
```

Olvasd be a `forras61.be` állományban található adatokat (ezek szavak, az első sor a darabszámot tartalmazza), és írd ki a képernyőre a beolvasott szavak közül a leghosszabbat!

```
using System;
using System.IO;
namespace szavak61
{
    class Program
    {
        static void Main(string[] args)
        {
            FileStream f = new FileStream("forras61.be", FileMode.Open);
            StreamReader r = new StreamReader(f);
            int db = Convert.ToInt32(r.ReadLine());
            string szo;
            int maxhossz = 0;
            string maxszo = "";
            for (int i = 0; i < db; i++)
            {
                szo = r.ReadLine();
                if (szo.Length >= maxhossz)
                {
                    maxhossz = szo.Length;
                    maxszo = szo;
                }
            }
            r.Close();
            f.Close();
            Console.WriteLine("A leghosszabb szó: " + maxszo);
            Console.ReadKey();
        }
    }
}
```