

# Egyszámjáték

Az egyszámjáték Mérő László matematikus találmánya. A játék nagyon egyszerű. Mindenkinek, aki a játék egy fordulójában részt kíván venni, tippelnie kell egy számra 1 és 99 között. A játékot az nyeri, aki a legkisebb olyan számra tippelt, amelyre csak ő tippelt egyedül, ha nincs ilyen szám, akkor a fordulónak nincs nyertese. Ebben a feladatban egy többfordulós egyszámjátékkal kapcsolatban kell feladatokat megoldania.

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.*
- *A megoldását úgy készítse el, hogy az azonos szerkezetű, de tetszőleges bemeneti adatok mellett is helyes eredményt adjon!*

Az `egyszamjatek.txt` állomány soronként tartalmazza a játékban részt vevő játékosok nevét és a fordulónként leadott tippjeit. A játékosok és a fordulók száma 5–10 közötti lehet. A tippek 1–99 közötti egész számok lehetnek. Az adatokat a szóköz karakter választja el egymástól. Az állományban nincs két egyforma nevű játékos.

Például: 3 12 1 8 5 8 1 2 1 4 Marci

A példában látható, hogy Marci tippjei a játék 10 fordulójában rendre 3 12 1 8 5 8 1 2 1 4 voltak.

1. Készítsen programot a következő feladatok megoldására, amelynek a forráskódját

`egyszamjatek` néven mentse el!

2. Olvassa be az `egyszamjatek.txt` állományban lévő adatokat és tárolja el egy olyan adatszerkezetben, ami a további feladatok megoldására alkalmas!

3. Határozza meg és írja ki a képernyőre a minta szerint, hogy a játékban hány játékos vett részt!

4. Határozza meg és írja ki a képernyőre a minta szerint, hogy a játékban hány fordulót játszottak a játékosok! Feltételezheti, hogy minden játékos minden fordulóban részt vett.

5. Döntse el és írja ki a képernyőre a minták szerint, hogy az első fordulóban tippelt-e valaki az 1-es számra!

6. Határozza meg és írja ki a minta szerint, hogy a fordulók során melyik volt a legnagyobb tipp!

7. Kérje be egy forduló sorszámát! Az adatbevitel előtt jelenjen meg a lehetséges legkisebb és legnagyobb fordulósorszám értéke! Például: „7. feladat: Kérem a forduló sorszámát [1–10]:” (Ebben az esetben 10 db forduló volt az `egyszamjatek.txt` állományban.) Ha a beadott sorszám nem felel meg a lehetséges értékeknek, akkor az 1. fordulóval dolgozzon a következő feladatokban!

8. Az előző feladatban bekért fordulóban határozza meg és írja ki a minta szerint a nyertes tipp értékét! Ha nem volt nyertes tipp a vizsgált fordulóban, akkor a „Nem volt egyedi tipp a megadott fordulóban!” szöveget jelenítse meg!

9. A 7. feladatban bekért fordulóban határozza meg és írja ki a minta szerint a nyertes játékos nevét! Ha nem volt nyertes a megadott fordulóban, akkor a „Nem volt nyertes a megadott fordulóban!” szöveget jelenítse meg!

10. Ha volt nyertes a 7. feladatban megadott fordulóban, akkor a minta szerint írja ki a nyertes forduló adatait a `nyertes.txt` állományba!

**MINTA A FELADATHOZ:**

3. feladat: Játékosok száma: 9

4. feladat: Fordulók száma: 10

5. feladat: Az első fordulóban volt egyes tipp!

6. feladat: A legnagyobb tipp a fordulók során: 13

7. feladat: Kérem a forduló sorszámát [1–10]: 3

8. feladat: A nyertes tipp a megadott fordulóban: 3

9. feladat: A megadott forduló nyertese: Andi

**MINTA nyertes.txt ÁLLOMÁNYHOZ:**

**Forduló sorszám: 2.**

**Nyertes tipp: 3**

**Nyertes játékos: Tibi**

Megoldás:

```
using System;
using System.IO;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace egyszamjatek
{
    class jatekos
    {
        public string nev { get; private set; }
        public List<int> tippek { get; private set; }
        public int fordulokszama { get { return tippek.Count - 1; } }
        public jatekos(string sor)
        {
            string[] m = sor.Split();
            nev = m[m.Length - 1];
            tippek = new List<int>();
            tippek.Add(-1); //a forduló sorszáma egyenlő lesz az indexxel
            for(int i = 0; i < m.Length - 1; i++)
            {
                tippek.Add(int.Parse(m[i]));
            }
        }
    }
    class Program
    {
        static void Main(string[] args)
        {
            //2.feladat
            List<jatekos> t = new List<jatekos>();
            foreach (var i in File.ReadAllLines("egyszamjatek.txt"))
            {
                t.Add(new jatekos(i));
                Console.WriteLine(" {0} ",i); //sorok kiírása
            }
            //3. feladat: a játékban hány játékos vett részt
            Console.WriteLine("3. feladat: játékosok száma: {0}",t.Count());
            //4. feladat: hány fordulót játszottak a játékosok
            Console.WriteLine("4. feladat: fordulók száma: {0} ", t[0].fordulokszama);
            //5. feladat Az első fordulóban volt-e 1-es tipp
            bool voltegyes = false;
            foreach(var i in t)
            {
                if (i.tippek[1] == 1)
                {
                    voltegyes = true;
                    break;
                }
            }
            Console.WriteLine("5. feladat: az első fordulóban {0} volt egyes tipp!", voltegyes?"":"nem");
            //6. feladat legnagyobb tipp
            int max = 0;
            foreach (var i in t)
            {
                foreach(var j in i.tippek)
                {
                    if (j > max) max = j;
                }
            }
            Console.WriteLine("6. feladat: legnagyobb tipp a fordulók során: {0}", max);
            //7. feladat: forduló sorszámanak bekérése (N)
            Console.Write("7. feladat: Kérem a forduló sorszámát [1-{0}]: ",t[0].fordulokszama);
            int fordulosorszama = int.Parse(Console.ReadLine());
            //Ha a beadott sorszám nem felel meg a lehetséges értékeknek akkor 1-el számolunk
            if (fordulosorszama < 1 || fordulosorszama > t[0].fordulokszama) fordulosorszama = 1;
            //8. feladat: melyik volt a nyertes tipp az N. fordulóban
            int[] stat = new int[100];
        }
    }
}
```

```

foreach(var i in t)
{
    stat[i.tippek[fordulosorszama]]++;
}
int nyertestipp = -1;
for(int i = 1; i < stat.Length; i++)
{
    if (stat[i] == 1)
    {
        nyertestipp = i;
        break;
    }
}
if (nyertestipp != -1) Console.WriteLine("8. feladat: A nyertes tipp a megadott
fordulóban: {0}",nyertestipp);
else Console.WriteLine("8. feladat: A megadott fordulóban nem volt egyedi tipp ");
//9. feladat nyertes játékos neve az N. fordulón
string fordulonyertese = "";
if (nyertestipp != -1)
{
    foreach(var i in t)
    {
        if (i.tippek[fordulosorszama] == nyertestipp)
        {
            fordulonyertese = i.nev;
            break;
        }
    }
    Console.WriteLine("9. feladat: A megadott forduló nyertese: {0} ", fordulonyertese);
}
else
    Console.WriteLine("9. feladat: Nem volt nyertes a megadott fordulóban ");
//9. feladat nyertes.txt
FileStream fnev = new FileStream("nyertes.txt", FileMode.Append);
StreamWriter fajlbairo = new StreamWriter(fnev);
if (nyertestipp != -1)
    {
        fajlbairo.WriteLine("Forduló sorszáma: {0}.", fordulosorszama);
        fajlbairo.WriteLine("Nyertes tipp: {0}.", nyertestipp);
        fajlbairo.WriteLine("Nyertes játékos: {0}.", fordulonyertese);    }
fajlbairo.Close();
fnev.Close();
Console.ReadKey();
}    }}

```