**C# feladatok gyűjteménye**

# Készítette:

Fehérvári Károly I6YF6E

Informatika tanár ma levelező tagozat

1. **Feladat: ALAPMŰVELETEK**

Készítsünk programot, amely bekér két egész számot. Majd kiszámolja a két szám összegét, különbségét, szorzatát és hányadosát.

### *Megoldás:*

static void Main(string[] args)

{

}

## Kiegész ítés:

int szam1; int szam2; int osszeg;

int kulonbseg; int szorzat; int hanyados;

Console.WriteLine("Alapműveletek program"); Console.Write("Kérem az első számot: "); szam1 = int.Parse(Console.ReadLine()); Console.Write("Kérem a második számot: "); szam2 = int.Parse(Console.ReadLine()); osszeg = szam1 + szam2;

kulonbseg = szam1 - szam2; szorzat = szam1 \* szam2; hanyados = szam1 / szam2;

Console.WriteLine("A két szám összege: {0}", osszeg); Console.WriteLine("A két szám különbsége: {0}", kulonbseg); Console.WriteLine("A két szám szorzata: {0}", szorzat); Console.WriteLine("A két szám hányadosa: {0}", hanyados); Console.ReadKey();

## Alakítsuk át úgy a programot, hogy a nagyobb számból vonja ki a kisebbet, valamint a nagyobb számot ossza el a kisebbel!

### *Megoldás:*

static void Main(string[] args)

{

int szam1; int szam2; int osszeg;

int kulonbseg; int szorzat; int hanyados;

Console.WriteLine("Alapműveletek program"); Console.Write("Kérem az első számot: "); szam1 = int.Parse(Console.ReadLine()); Console.Write("Kérem a második számot: "); szam2 = int.Parse(Console.ReadLine()); osszeg = szam1 + szam2;

szorzat = szam1 \* szam2; if (szam1 > szam2)

{

kulonbseg = szam1 - szam2; hanyados = szam1 / szam2;

}

else

{

kulonbseg = szam2 - szam1; hanyados = szam2 / szam1;

};

Console.WriteLine("A két szám összege: {0}", osszeg); Console.WriteLine("A két szám különbsége: {0}", kulonbseg); Console.WriteLine("A két szám szorzata: {0}", szorzat); Console.WriteLine("A két szám hányadosa: {0}", hanyados); Console.ReadKey();

}

# Feladat: PÁROSSÁG VIZSGÁLATA

## Készítsünk programot, mely bekér a felhasználótól egy számot, majd kiírja az adott számról, hogy páros, páratlan, vagy nulla.

### *Megoldás:*

static void Main(string[] args)

{

int szam;

Console.WriteLine("Párosság vizsgáló program"); Console.Write("Kérek egy számot: ");

szam = int.Parse(Console.ReadLine());

if (szam % 2 == 0) Console.WriteLine("A szám páros");

if (szam % 2 != 0) Console.WriteLine("A szám páratlan"); if (szam == 0) Console.WriteLine("A szám értéke nulla"); Console.ReadKey();

}

# Feladat: OSZTÓ VIZSGÁLAT

## Készítsünk programot, amely bekér a felhasználótól két számot. A program döntse el, hogy az elsőként bekért számnak osztója-e a másodikként bekért szám. Amennyiben igen, írja ki az osztás eredményét is.

### *Megoldás:*

static void Main(string[] args)

{

int szam; int oszto; int ered;

Console.WriteLine("Osztója-e?"); Console.Write("Kérek egy számot: "); szam = int.Parse(Console.ReadLine()); Console.Write("Kérek egy számot: "); oszto = int.Parse(Console.ReadLine()); if (szam % oszto == 0)

{

ered = szam / oszto;

Console.WriteLine("Az {0} osztója {1}-nek.", oszto, szam ); Console.WriteLine("{0} / {1} = {2}", szam, oszto, ered);

}

else Console.WriteLine("A(z) {0} NEM osztója {1}-nek.", oszto, szam); Console.ReadKey();

}

# Feladat: HÁROMSZÖG S ZERKES ZT HETŐSÉG

## Készítsünk programot, mely eldönti egy háromszög három oldala alapján, hogy az adott háromszög szerkeszthető-e.

### *Megoldás:*

static void Main(string[] args)

{

int a; int b; int c;

Console.WriteLine("Háromszög szerkeszthetőség"); Console.WriteLine("Kérem a hátomszög oldalait!"); Console.Write("a: ");

a = int.Parse(Console.ReadLine()); Console.Write("b: ");

b = int.Parse(Console.ReadLine()); Console.Write("c: ");

c = int.Parse(Console.ReadLine());

if ((a + b > c) && (a + c > b) && (b + c > a)) Console.WriteLine("A háromszög szerkeszthető");

else Console.WriteLine("A háromszög nem szerkeszthető"); Console.ReadKey();

}

# Feladat: SZÍNES KOR

## Készítsünk programot, mely a következő feladatokat végzi el!

Bekéri a felhasználó nevét, valamint születésének évét. Kiszámítja a felhasználó életkorát. Ezután egy üres képernyőn, melynek kék a háttere, a képernyő közepén megjeleníti sárga színnel a felhasználó nevét, majd alatta piros színnel a felhasználó életkorát.

### *Megoldás:*

static void Main(string[] args)

{

string nev; int ev;

int kora; string korastr;

Console.WriteLine("Névjegy"); Console.Write("Kérem a nevet: "); nev = Console.ReadLine();

Console.Write("Kérem a születési évet: "); ev = int.Parse(Console.ReadLine());

kora = DateTime.Today.Year - ev; korastr = Convert.ToString(kora);

Console.BackgroundColor=ConsoleColor.Blue; Console.Clear();

Console.ForegroundColor = ConsoleColor.Yellow; Console.SetCursorPosition((Console.WindowWidth / 2)-(nev.Length/2),

Console.WindowHeight / 2);

Console.WriteLine(nev); Console.ForegroundColor = ConsoleColor.Red;

Console.SetCursorPosition((Console.WindowWidth / 2)- (korastr.Length/2), (Console.WindowHeight / 2)+1);

Console.WriteLine(kora); Console.ReadKey();

}

# Feladat: SZÁMOK 1 -N-IG

## Készítsünk programot, amely bekér egy N számot és kiírja a számokat 1-től N-ig.

### *Megoldás:*

static void Main(string[] args)

{

}

## Kiegész ítés:

int N; int i;

Console.WriteLine("Számok 1-N-ig");

Console.Write("Kérek egy tetszőleges pozitív egész számot: "); N = int.Parse(Console.ReadLine());

for (i = 1; i <= N; i++) Console.Write("{0}, ", i); Console.ReadKey();

## Módosítsuk úgy a programot, hogy az csak a páros számokat írja ki!

### *Megoldás:*

static void Main(string[] args)

{

}

### *Megjegyzés:*

int N; int i;

Console.WriteLine("Számok 1-N-ig");

Console.Write("Kérek egy tetszőleges pozitív egész számot: "); N = int.Parse(Console.ReadLine());

for (i = 1; i <= N; i++)

if (i % 2 == 0) Console.Write("{0}, ", i); Console.ReadKey();

## A feladatban akár a páratlan számok kiíratása, akár a tetszőleges számmal történő oszthatóság vizsgálata lehetséges, mint kiegészítés.

1. **Feladat: PRÍMSZÁM**

Készítsünk programot, mely a következő feladatot látja el:

Bekér a felhasználótól egy számot, majd eldönti róla, hogy az adott szám prímszám-e.

### *Megoldás:*

static void Main(string[] args)

{

int szam; int i;

int osztok;

Console.Write("Kérem a vizsgálandó számot: "); szam = int.Parse(Console.ReadLine());

osztok = 0;

for (i = 2; i <= szam / 2; i = i + 1)

{

if (szam % i == 0) osztok = osztok + 1;

}

}

## Kiegész ítés:

if (osztok == 0) Console.WriteLine("A megadott szám PRÍM"); else Console.WriteLine("A megadott szám NEM PRÍM"); Console.ReadKey();

## A program írja ki, hogy a felhasználó által megadott szám osztóit, illetve, hogy az adott számnak hány osztója van egyen és önmagán kívül.

### *Megoldás:*

static void Main(string[] args)

{

int szam; int i;

int osztok;

Console.Write("Kérem a vizsgálandó számot: "); szam = int.Parse(Console.ReadLine());

osztok = 0;

for (i = 2; i <= szam / 2; i = i + 1)

{

if (szam % i == 0)

{

osztok = osztok + 1; Console.WriteLine(i);

}

}

if (osztok == 0) Console.WriteLine("A megadott szám PRÍM"); else

{

Console.WriteLine("A megadott szám NEM PRÍM"); Console.Write("A szám TERMÉSZETES osztóinak száma: "); Console.WriteLine(osztok);

}

}

### *Megjegyzés:*

Console.ReadKey();

## A feladat további kiegészítéseként kiírathatjuk a szám osztóit is.

1. **Feladat: PRÍMEK 1-N-IG**

Készítsünk programot, mely a következő feladatot látja el:

Bekér a felhasználótól egy számot, majd a megadott számig kiírja a prímszámokat.

### *Megoldás:*

static void Main(string[] args)

{

int szam; int i;

int osztok; int N;

Console.Write("Kérem a vizsgálandó végértéket: "); N = int.Parse(Console.ReadLine());

for (szam = 2; szam <= N; szam = szam + 1)

{

osztok = 0;

for (i = 2; i <= szam / 2; i = i + 1)

{

if (szam % i == 0) osztok = osztok + 1;

}

if (osztok == 0) Console.WriteLine(szam);

}

Console.ReadKey();

}

1. **Feladat: PRÍME TÉNYE ZŐS FELBONTÁS** Készítsünk programot, mely elvégzi egy szám prímtényezős felbontását. *Megoldás:*

static void Main(string[] args)

{

int szam; int i;

int osztok; int veg;

Console.Write("Kérem a vizsgálandó végértéket: "); veg = int.Parse(Console.ReadLine());

szam = 1; do

{

szam=szam+1; osztok = 0;

for (i = 2; i <= szam % 2; i = i + 1)

{

if (szam % i == 0) osztok = osztok + 1;

}

if (osztok == 0)

{

if (veg % szam == 0)

{

Console.WriteLine(szam); veg = veg / szam;

szam = szam - 1;

}

}

}

while (szam<=veg); Console.ReadKey();

}

1. **Feladat: FAKTORIÁLIS SZÁMÍTÁS** Készítsünk programot, mely kiszámítja egy szám faktoriálisát. *Megoldás:*

static void Main(string[] args)

{

int szam; int fakt; int i;

Console.WriteLine("Faktoriális program"); szam = int.Parse(Console.ReadLine()); fakt = 1;

for (i=1; i<=szam;i++) fakt = fakt\*i;

Console.WriteLine("A megadott szám faktoriálisa: {0}", fakt); Console.ReadKey();

}

# Feladat: ELDÖNTÉS-KIVÁLASZTÁS

## Készítsünk programot, mely a következő feladatokat végzi el:

* + Feltölt egy N elemű tömböt egy és száz közé eső véletlen számokkal. A tömb méretét a felhasználó határozza meg!
  + Feltöltés után a tömb elemeit a program írja ki a képernyőre!
  + Döntse el a program, hogy a tömb tartalmaz-e páros elemet! (eldöntés tétele)
  + Határozza meg az első páros szám sorszámát! (kiválasztás tétele)
  + A program négy, jól elkülöníthető, egymástól független részből álljon!

Figyelem: A tömb elemeinek sorszámozása 0-val kezdődik!!! Tehát, az egy elemű tömb egyetlen elemének sorszáma 0.

### *Megoldás:*

static void Main(string[] args)

{

Console.WriteLine("Eldöntés tétele, kiválasztás tétele"); Random rnd = new Random();

Console.Write("Hány eleme legyen a tömbnek? "); int N = int.Parse(Console.ReadLine());

int[] A = new int[N];

//feltöltés

for(int i=0;i<N;i++)

{

A[i] = rnd.Next(100) + 1;

}

//kiiratás

foreach (int j in A)

{

Console.Write("{0}, ", j);

}

//eldöntés int cv = 0;

bool vane = false; do

{

if (A[cv] % 2 ==0) vane = true; cv = cv + 1;

}

while ((cv < N)&&(vane==false));

if (vane == true) Console.WriteLine("A tömb tartalmaz páros elemet"); else Console.WriteLine("A tömb nem tartalmaz páros elemet");

//kiválasztás cv = 0;

int sorsz = -1; do

{

if (A[cv] % 2 == 0) sorsz = cv; cv = cv + 1;

}

while ((cv < N) && (sorsz == -1)); if (sorsz != -1)

Console.WriteLine("A tömb első páros elemének helye {0}", sorsz); Console.ReadLine();

}

# Feladat: KIVÁLOGATÁS-MEGSZÁMLÁLÁS

## Készítsünk programot, mely a következő feladatokat végzi el:

* + Feltölt egy N elemű tömböt egy és száz közé eső véletlen számokkal. A tömb méretét a felhasználó határozza meg!
  + Feltöltés után a tömb elemeit a program írja ki a képernyőre!
  + Másolja át a program egy másik tömbbe a páros számokat! (kiválogatás tétele)
  + Számolja meg a program, hogy hány páros számot tartalmazott a tömb! (megszámlálás tétele)
  + Írja ki a program a páros elemek számát!
  + Írja ki a program a páros elemeket!

Figyelem: A tömb elemeinek sorszámozása 0-val kezdődik!!! Tehát, az egy elemű tömb egyetlen elemének sorszáma 0.

### *Megoldás:*

static void Main(string[] args)

{

Console.WriteLine("Eldöntés tétele, kiválasztás tétele"); Random rnd = new Random();

Console.Write("Hány eleme legyen a tömbnek? "); int N = int.Parse(Console.ReadLine());

int[] A = new int[N];

//feltöltés

for(int i=0;i<N;i++)

{

A[i] = rnd.Next(100) + 1;

}

//kiiratás

foreach (int j in A)

{

Console.Write("{0}, ", j);

}

//Kiválogatás-megszámlálás int cv = 0;

int [] paros = new int[N]; int db = 0;

for (cv = 0; cv<N;cv++)

{

if (A[cv] % 2 == 0)

{

db=db+1; paros[db]=A[cv];

}

}

//Kiiratás Console.WriteLine();

Console.WriteLine("A tömbben {0} darab páros szám található", db); for (int i=1;i<=db;i++) Console.Write("{0}, ", paros[i]); Console.ReadLine();

}

# Feladat: SZÉTVÁL OGATÁS

## Készítsünk programot, mely a következő feladatokat végzi el:

* + Feltölt egy N elemű tömböt egy és száz közé eső véletlen számokkal. A tömb méretét a felhasználó határozza meg!
  + Feltöltés után a tömb elemeit a program írja ki a képernyőre!
  + Másolja át a program egy másik tömbbe a páros számokat!
  + Másolja át a program egy harmadik tömbbe a páratlan számokat!
  + Számolja meg a program, hogy hány páros illetve hány páratlan számot tartalmazott a tömb! (megszámlálás tétele)
  + Írja ki a program a páros illetve páratlan elemek számát!
  + Írja ki a program a páros illetve páratlan elemeket!

Figyelem: A tömb elemeinek sorszámozása 0-val kezdődik!!! Tehát, az egy elemű tömb egyetlen elemének sorszáma 0.

### *Megoldás:*

static void Main(string[] args)

{

Console.WriteLine("Eldöntés tétele, kiválasztás tétele"); Random rnd = new Random();

Console.Write("Hány eleme legyen a tömbnek? "); int N = int.Parse(Console.ReadLine());

int[] A = new int[N];

//feltöltés

for(int i=0;i<N;i++)

{

A[i] = rnd.Next(100) + 1;

}

//kiiratás

foreach (int j in A)

{

Console.Write("{0}, ", j);

}

//Kiválogatás-megszámlálás int cv = 0;

int [] paros = new int[N]; int [] paratlan = new int[N]; int parosdb = 0;

int paratlandb = 0;

for (cv = 0; cv<N;cv++)

{

if (A[cv] % 2 == 0)

{

parosdb = parosdb + 1; paros[parosdb] = A[cv];

}

else

{

paratlandb = paratlandb + 1; paratlan[paratlandb] = A[cv];

}

}

parosdb);

paratlandb);

//Kiiratás Console.WriteLine();

Console.WriteLine("A tömbben {0} darab páros szám található",

for (int i=1;i<=parosdb;i++) Console.Write("{0}, ", paros[i]); Console.WriteLine();

Console.WriteLine("A tömbben {0} darab páratlan szám található",

for (int i = 1; i <= paratlandb; i++) Console.Write("{0}, ",

paratlan[i]);

Console.ReadLine();

}

# Feladat: RENDEZÉ S

## Készítsünk programot, mely a következő feladatokat végzi el:

* + Feltölt egy N elemű tömböt egy és száz közé eső véletlen számokkal. A tömb méretét a felhasználó határozza meg!
  + Feltöltés után a tömb elemeit a program írja ki a képernyőre!
  + Rendezze a program a tömb elemeit emelkedő sorrendbe!
  + Írja ki a program a rendezett tömb elemeit!

### *Megoldás:*

static void Main(string[] args)

{

Console.WriteLine("Eldöntés tétele, kiválasztás tétele"); Random rnd = new Random();

Console.Write("Hány eleme legyen a tömbnek? "); int N = int.Parse(Console.ReadLine());

int[] A = new int[N];

//feltöltés

for (int i = 0; i < N; i++)

{

A[i] = rnd.Next(100) + 1;

}

//kiiratás

foreach (int j in A)

{

Console.Write("{0}, ", j);

}

//Rendezés

for (int i = 0; i < N-1; i++)

{

for (int j = i+1; j < N; j++)

{

if (A[j] < A[i])

{

int c = A[i]; A[i] = A[j]; A[j] = c;

}

}

}

Console.WriteLine();

Console.WriteLine("A tömb elemei emelkedő sorrendben: "); for (int i = 0; i < N; i++) Console.Write("{0}, ", A[i]); Console.ReadLine();

}

# Feladat: LISTA

## Készítsünk programot, mely a következő feladatokat végzi el:

* + Feltölt egy N elemű listát egy és száz közé eső véletlen számokkal. A lista méretét a program elején egy konstansban határozzuk meg!
  + Feltöltés után a lista elemeit a program írja ki a képernyőre!
  + Írjuk ki a képernyőre a lista legkisebb elemét!
  + Írjuk ki a képernyőre a lista legnagyobb elemét!
  + Írjuk ki a képernyőre a lista elemeinek összegét!
  + Írjuk ki a képernyőre a lista elemeinek átlagát!

A program elkészítése során éljünk a lista adatszerkezet nyújtotta lehetőségekkel!

### *Megoldás:*

static void Main(string[] args)

{

const int elemszam = 20; Random rnd = new Random();

List<int> szamok = new List<int>();

//Lista feltöltése

for (int i = 1; i < elemszam; i++)

{

szamok.Add(rnd.Next(100) + 1);

}

//Lista kiíratása

Console.WriteLine("A Lista elemei: ");

foreach (int i in szamok) Console.Write("{0}, ", i);

//Legkisebb elem Console.WriteLine();

Console.WriteLine("A Lista legkisebb eleme: {0}", szamok.Min());

//Legnagyobb elem Console.WriteLine();

Console.WriteLine("A Lista nagyobb eleme: {0}", szamok.Max());

//Lista elemeinek összege Console.WriteLine();

Console.WriteLine("A Lista elemeinek összege: {0}", szamok.Sum());

//Lista átlaga Console.WriteLine();

Console.WriteLine("A Lista átlaga: {0}", szamok.Average()); Console.ReadKey();

}

# Feladat: MÁTRIX

## Készítsünk programot, mely a következő feladatokat végzi el:

* + Feltölt egy n\*m elemű mátrixot 1-99 közé eső véletlen számokkal. A program elején a sorok és oszlopok számát konstansok tárolják. (m, n)
  + Feltöltés után a mátrix elemeit a program írja ki a képernyőre, rendezetten, átláthatóan.
  + Képezze a program a mátrix valamely tükörmátrixát!
  + A kész tükörmátrixot, az eredeti mátrix mellé, hasonló módon rendezetten írja ki a program!
  + A programban végrehajtott tevékenységeket válasszuk el egymástól!

### *Megoldás:*

static void Main(string[] args)

{

const int m = 6; const int n = 6;

int[,] matrix = new int[m, n]; Random rnd = new Random();

// mátrix feltöltése adatokkal for (int i = 0; i < m; i++)

{

for (int j = 0; j < n; j++)

{

matrix[i, j] = rnd.Next(99)+1;

}

}

// mátrix kiírása

Console.WriteLine("Az eredeti mátrix: "); for (int i = 0; i < m; i++)

{

for (int j = 0; j < n; j++)

{

Console.SetCursorPosition(i \* 3,5+(j \* 2)); Console.Write(matrix[i, j]);

}

}

// A mátrix tükrözése

for (int i = 0; i < m; i++)

{

for (int j = i; j < n; j++)

{

int cs = matrix[i, j]; matrix[i, j] = matrix[j, i]; matrix[j, i] = cs;

}

}

// Tükörmátrix kiiratása Console.SetCursorPosition(40, 0); Console.Write("Az eredeti mátrix: "); for (int i = 0; i < m; i++)

{

for (int j = 0; j < n; j++)

{

Console.SetCursorPosition(40+(i \* 3),5+(j \* 2)); Console.Write(matrix[i, j]);

}

}

Console.ReadKey();

}

# Feladat: LOTTÓ

## Készítsünk programot, mely a következő feladatokat végzi el:

* + Feltölt egy 5 elemű tömböt egy és 90 közé eső véletlen számokkal. Ez a tömb a nyerőszámokat tartalmazó tömb.
  + Bekér egy 5 elemű tömböt a felhasználótól. A számok itt is 1 és 90 közé eshetnek. (ezt a programnak figyelnie is kell, valamint azt is, hogy a megadott szám 1 és 90 közé essen). Ez a tömb a felhasználó által megadott tippeket tartalmazza.
  + Feltöltés után a tömb elemeit a program írja ki a képernyőre!
  + Rendezze a program a két tömb elemeit emelkedő sorrendbe!
  + Vizsgálja meg a program, hogy hány találatot ért el a felhasználó, majd írja ki azt.
  + Írja ki a program emelkedő sorrendben a felhasználó által adott tippeket illetve a nyerő lottószámokat is.

### *Megoldás:*

static void Main(string[] args)

{

Random rnd = new Random();

//Nyerő számok generálása int[] nyero = new int[5]; int cv = 0;

while (cv < 5)

{

int aktszam = rnd.Next(90) + 1; bool vane = false;

for (int i = 0; i<cv; i++) if (nyero[i] == aktszam) vane = true; if (vane == false)

{

nyero[cv] = aktszam; cv = cv + 1;

}

}

//Felhasználó számainak bekérése int[] felh = new int[5];

Console.WriteLine("Kérem adja meg a tippeket!"); cv = 0;

while (cv < 5)

{

Console.WriteLine();

Console.Write("Kérem a {0}. számot: ", cv+1); int aktszam = int.Parse(Console.ReadLine()); bool vane = false;

for (int i = 0; i < cv; i++) if (felh[i] == aktszam) vane = true; if (vane == false)

{

if ((aktszam >= 1) && (aktszam <= 90))

{

felh[cv] = aktszam; cv = cv + 1;

}

Lottószám!");

}

else Console.WriteLine("A megadott szám nem megfelelő

else Console.WriteLine("Ezt a számot már megadta egyszer!");

}

//Nyerő számok rendezése

for (int i = 0; i < 5 - 1; i++)

{

for (int j = i + 1; j < 5; j++)

{

if (nyero[j] < nyero[i])

{

int c = nyero[i]; nyero[i] = nyero[j]; nyero[j] = c;

}

}

}

//Felhasználó számainak rendezése for (int i = 0; i < 5 - 1; i++)

{

for (int j = i + 1; j < 5; j++)

{

if (felh[j] < felh[i])

{

int c = felh[i]; felh[i] = felh[j]; felh[j] = c;

}

}

}

//Találatok keresése int talalat = 0;

for (int i = 0; i < 5; i++)

{

for (int j = 0; j < 5; j++)

{

if (felh[i] == nyero[j]) talalat = talalat + 1;

}

}

//Kiiratás Console.WriteLine();

Console.WriteLine("A találatok száma: {0}", talalat); Console.WriteLine("A nyrő számok: ");

foreach (int i in nyero) Console.Write("{0}, ", i); Console.WriteLine();

Console.WriteLine("A felhasználó számai: "); foreach (int i in felh) Console.Write("{0}, ", i); Console.ReadKey();

}

# Feladat: SZÁMRE NDS ZEREK – ÁTVÁLTÁS 10 -X

## Készítsünk programot, mely a következő feladatokat végzi el:

* + Bekér a program egy decimális számrendszerbeli számot.
  + Bekéri a program, hogy mely számrendszerbe kell a számot átváltani
  + Végrehajtja az átváltást.
  + Kiírja az eredményt a kért számrendszerben.

### *Megoldás:*

static void Main(string[] args)

{

int i;

Console.Write("Kérem az átváltandó számot: "); int szam = int.Parse(Console.ReadLine());

//A feladott szám Console.WriteLine();

Console.Write("Milyen számrendszerbe váltsam át? (1-10,16)"); int szr = int.Parse(Console.ReadLine());

Console.WriteLine();

//A feladott számrendszer int[] eredszam = new int[16];

string eredszamtxt = string.Empty;

//A kiszámíott végleges eredmény tárolása string betu = string.Empty;

//Egy adott érték, amit a fentebb írt txt-hez adunk int ered;

int marad;

for (int j = 1; j <= 5; j++)

{

ered = szam; i = 0;

eredszamtxt = string.Empty; do

{

i = i + 1;

marad = ered % szr; ered = ered / szr; eredszam[i] = marad;

}

while (ered != 0); int hossz = i;

for (i = hossz; i != 0; i = i - 1)

{

if ((szr == 16) && (eredszam[i] >= 10))

{

if (eredszam[i] == 10) betu = "A"; if (eredszam[i] == 11) betu = "B"; if (eredszam[i] == 12) betu = "C"; if (eredszam[i] == 13) betu = "D"; if (eredszam[i] == 14) betu = "E"; if (eredszam[i] == 15) betu = "F";

}

else betu = Convert.ToString(eredszam[i]); eredszamtxt = eredszamtxt + betu;

}

}

Console.WriteLine("Az eredmény {0} szr-ben: {1}", szr, eredszamtxt); Console.ReadKey();

}

# Feladat: SZÁMRE NDS ZEREK – ÁTVÁLTÁS X-10

## Készítsünk programot, mely a következő feladatokat végzi el:

* + Bekér a program egy tetszőleges számrendszerbeli számot.
  + Bekéri a program, hogy az adott szám mely számrendszerben lett megadva.
  + Átváltja a megadott számot decimális számrendszerbe.
  + Kiírja az eredményt.

### *Megoldás:*

static void Main(string[] args)

{

//16-os számrendszerre nem működik... int i;

Console.Write("Kérem az átváltandó számot: "); string szam = Console.ReadLine(); Console.WriteLine();

Console.Write("Milyen számrendszerben van a megadott szám? "); int szr=int.Parse(Console.ReadLine());

int[] szamjegy = new int[11]; string aktszamjegy;

int szamhossz = szam.Length;

//Megjegyzi, hogy hány karakter az átváltandó szám for (i = 1; i != szamhossz + 1; i = i + 1)

{

aktszamjegy = szam.Substring(i - 1, 1); szamjegy[i] = int.Parse(aktszamjegy);

}

int helyiertek = 1; int ered = 0;

for (int j = szamhossz; j != 0; j = j - 1)

{

ered = ered + (szamjegy[j] \* helyiertek); helyiertek = helyiertek \* szr;

}

Console.WriteLine("A szám értéke tizes számrendszerben: {0}", ered); Console.ReadKey();

}

# Feladat: SZÁMRE NDS ZEREK – FÜG GVÉNYEK ALKALMAZÁSA

## Készítsünk programot, mely a következő feladatokat végzi el:

* + Bekér a program egy tetszőleges számrendszerbeli számot.
  + Bekéri a program az adott szám milyen számrendszerben lett megadva. (2-10-ig)
  + Bekéri a program, hogy milyen számrendszerbe váltsa át az adott számot? (2-10-ig, 16 is)
  + Átváltja a megadott számot a kért számrendszerbe. számrendszerbe.
  + Kiírja az eredményt.

A program elkészítéséhez függvényeket kell használni!

A program tudjon tetszőleges számrendszerből bármely más számrendszerbe átváltani.

### *Megoldás:*

class Program

{

public static string eredmeny;

static void Main(string[] args)

{

Console.Write("Kérem az átváltandó számot: "); string szam = Console.ReadLine(); Console.WriteLine();

Console.Write("Milyen számrendszerben van a megadott szám? "); int szr = int.Parse(Console.ReadLine());

Console.WriteLine();

Console.Write("Milyen számrendszerbe váltsam át a megadott számot? "); int feladatszr = int.Parse(Console.ReadLine());

if ((feladatszr == 10) && (szr <= 10)) eredmeny = ATX10(szam, szr);

if ((feladatszr != 10) && (szr == 10)) eredmeny = AT10X(szam, feladatszr);

if ((feladatszr != 10) && (szr != 10))

{

eredmeny = ATX10(szam, szr);

eredmeny = AT10X(eredmeny, feladatszr);

};

Console.WriteLine("A megadott {0} számrendszerbeli {1} szám érékte",szr ,szam);

Console.WriteLine("{0} számrendszerben: {1}", feladatszr, eredmeny); Console.ReadKey();

}

static string ATX10(string szam, int szr)

{

//16-os számrendszerre nem működik... int i;

int[] szamjegy = new int[11]; string aktszamjegy;

int szamhossz = szam.Length;

//Megjegyzi, hogy hány karakter az átváltandó szám for (i = 1; i != szamhossz + 1; i = i + 1)

{

aktszamjegy = szam.Substring(i - 1, 1); szamjegy[i] = int.Parse(aktszamjegy);

}

int helyiertek = 1; int ered = 0;

for (int j = szamhossz; j != 0; j = j - 1)

{

ered = ered + (szamjegy[j] \* helyiertek); helyiertek = helyiertek \* szr;

}

return Convert.ToString(ered);

}

static string AT10X(string szam, int szr)

{

int i;

int[] eredszam = new int[16]; string eredszamtxt = string.Empty string betu = string.Empty

int ered; int marad;

for (int j = 1; j <= 5; j++)

{

ered = int.Parse(szam); i = 0;

eredszamtxt = string.Empty; do

{

i = i + 1;

marad = ered % szr; ered = ered / szr; eredszam[i] = marad;

}

while (ered != 0); int hossz = i;

for (i = hossz; i != 0; i = i - 1)

{

if ((szr == 16) && (eredszam[i] >= 10))

{

if (eredszam[i] == 10) betu = "A"; if (eredszam[i] == 11) betu = "B"; if (eredszam[i] == 12) betu = "C"; if (eredszam[i] == 13) betu = "D"; if (eredszam[i] == 14) betu = "E"; if (eredszam[i] == 15) betu = "F";

}

else betu = Convert.ToString(eredszam[i]); eredszamtxt = eredszamtxt + betu;

}

}

return eredszamtxt;

}

}