



# C# .NET

LABORATOR 2

# Ce invatam azi



- ▶ Masive
- ▶ Exemplu practic pe o clasa
- ▶ Transmiterea parametrilor
- ▶ Interfete

# Masive



- ▶ Structuri de date omogene si continue
- ▶ In C#, masivele sunt tipuri referențiale derivate din clasa abstracta `System.Array` (crearea clasei derivate se face automat de catre compilator). `System.Array` este mostenita direct din `System.Object`
- ▶ Elementele masivelor pot fi de orice tip suportat (tipuri referentiale, tipuri valorice, alte masive)
- ▶ Sunt accesate prin intermediul indicilor (incepând cu 0)
- ▶ Dimensiunea masivelor este stabilita la crearea acestora (la rulare) si nu poate fi modificata pe parcurs. Limbajul suporta atat masive unidimensionale, cat si masive multidimensionale.

# Metode specifice



- ▶ Length: numarul total de elemente din masiv
- ▶ GetLength(dim): numarul de elemente de pe o dimensiune
- ▶ Rank: numarul de dimensiuni ale masivului
- ▶ Clone(): creeaza o copie a masivului; Atenție: in cazul tipurilor referentiale sunt copiate numai referintele, nu si obiectele referite
- ▶ Copy(), CopyTo(): copiaza secțiuni din masiv in alt masiv
- ▶ Multe altele... atat pe instante, cat si pe clasa

# Exemplu practic



- ▶ Clasa Persoana
- ▶ Vector de int cu id-urile persoanelor
- ▶ Il sortam, il parcurgem si afisam
- ▶ Il copiem intr-o matrice
- ▶ Facem o matrice in zig-zag
- ▶ Facem o lista de Persoane
- ▶ Copiem id-urile in ea
- ▶ Declaram un dictionar, punem persoane in functie de id-uri

# Transmiterea parametrilor



- ▶ Se transmit prin valoare
- ▶ Ref si out
- ▶ Params (vector si obligatoriu ultimul)
- ▶ Exemplu pe persoane, cu modificare/nemodificare a referintei
- ▶ Exemplu ref si out
- ▶ Suma unor intregi (params)

# ICollection si ICollection



- ▶ Metoda Clone() implicita
- ▶ Copie propriu-zisa