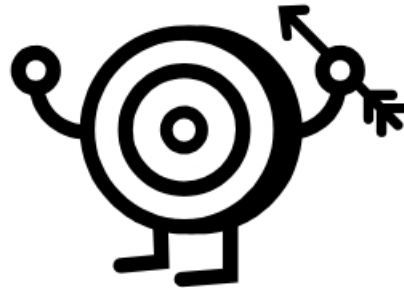


# Aula 05

## Array



# Objetivo da Aula



Desenvolver programas utilizando  
a  
estrutura de dados array.

# Agenda



- Array
  - Definição
  - Declaração
  - Acessando um elemento
  - Acessando todos os elementos
  - Classificação

# Array

Estrutura de dados composta de uma sequência estática de elementos, onde cada elemento é capaz de armazenar dados de um mesmo tipo, sendo identificado por índice(s) que indicam sua posição na estrutura.

<del>Variável Simples</del>
<del>SALARIO</del>
1.500,00

VETOR: Array Unidimensional	
SALÁRIO	
0	1500,00
1	3000,00
2	2500,00
3	2800,00
4	5000,00
5	4300,00

Índice

MATRIZ: Array Multidimensional				
NOTA				
	0	1	2	3
0	7,0	10,0	9,0	8,0
1	10,0	3,5	7,0	9,0
2	8,0	6,0	4,5	5,0
3	6,0	9,0	7,0	8,0

Índice das linhas

Índice das colunas

# Declaração de Array

```
double [] salario; // Apenas declarando um double [,] nota; // Apenas declarando uma MATRIZ
```

```
/* Criando um VETOR com 3 posições e cada elemento  
* é inicializado (new) com o valor padrão */
```

```
salario = new double[3];
```

```
/* Criando uma MATRIZ 2X3 (2 linhas e 3 colunas) e  
* cada elemento é inicializado (new) com o valor padrão */
```

```
nota = new double[2,3];
```

```
// Atribuindo valor salario[0]  
= 2;
```

```
nota[1, 2] = 4.5;
```

```
Console.WriteLine(salario[0]);  
Console.WriteLine(salario[1]);  
Console.WriteLine(nota[0, 0]);  
Console.WriteLine(nota[1, 2]);
```

## Valores Padrões para tipos primitivos

int	0
Double	0
String	""
Bool	false

Obs.: Tipos não primitivos inicializa com null

```
2  
0  
0  
4.5
```

# Declaração de Array

```
/* Declarando e criando um VETOR com 3 posições e  
* cada elemento é inicializado com o valor padrão */  
double [] salario = new double[3];
```

```
/* Declarando e criando uma MATRIZ 2,3 e  
* cada elemento é inicializado com o valor padrão */  
double[,] nota = new double[2, 3];
```

```
/* Declarando e criando um VETOR com 4 posições e  
* cada elemento é inicializado com um valor específico */  
  
string[] curso = {"ADS", "CSTDG", "ECP", "RC"};
```

```
/* Declarando e criando uma MATRIZ 2x3 e  
* cada elemento é inicializado com um valor específico */  
  
double [,] nota = { {1,2,3}, {4,5,6} };
```

# Acessando um Elemento do Array

Para acessar um elemento de um array, devemos indicar, entre colchetes, o número da posição desejada, referenciando seu(s) índice(s):

Exemplos:

```
salario[1] = Convert.ToDouble(Console.ReadLine());  
Console.WriteLine(salario[3]);  
Atual = salario[5];  
salario[5] = 1800.00;  
if (salario[3] > 3000) { ... }
```

```
nota[0,2] = Convert.ToDouble(Console.ReadLine());  
Console.WriteLine(nota[1,2]);  
Media = ((nota[0,0] + nota[0,1] + nota[0,2] + nota[0,3]) / 4;  
nota[3,0] = 6.0;  
if (nota[0,1] < 7) { ... }
```

# Acessando Todos os Elementos de um Array Unidimensional

Uma maneira de acessamos um array unidimensional é usar uma estrutura de repetição, onde a variável que controla a quantidade de iterações é usado como o índice do array.

## Exemplos:

Para somarmos todos os salários armazenados em um array de 6 posições

```
soma = 0;
for ( int i = 0; i < 6; i++)
{
    soma = soma + salario[i];
}
```



# Acessando Todos os Elementos de um Array Multidimensional

Uma maneira de acessamos um array multidimensional é usar uma estrutura de repetição para cada dimensão do array, onde a variável que controla a quantidade de iterações é usado como o índice de cada dimensão do array.

## Exemplos:

Somar todos os elementos armazenados em um array 4x5.

```
soma = 0;
for ( int i = 0; i < 4; i++)
{
    for ( int j = 0; j < 5; j++)
    {
        soma = soma + nota[i,j];
    }
}
```

# Classificação de Array Unidimensional


## Explicando a Técnica

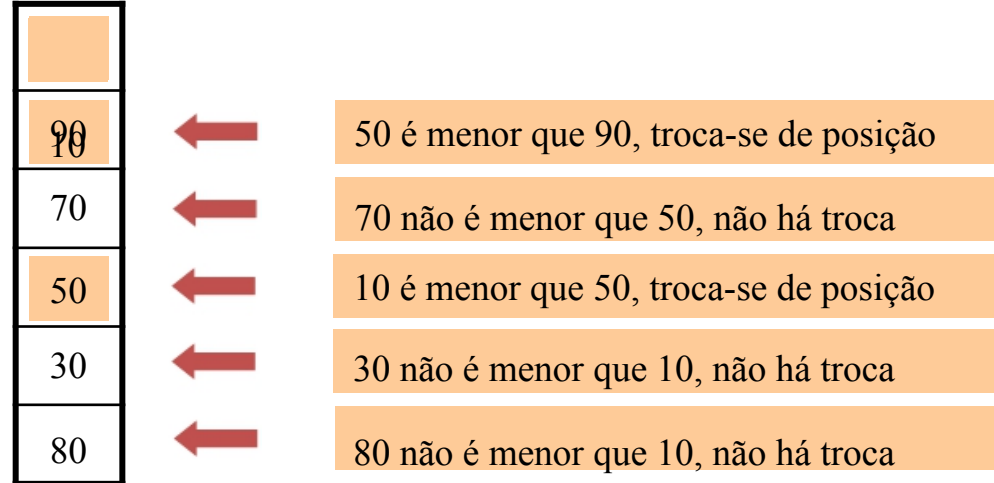
A técnica utilizada é a de comparar cada elemento do vetor com os demais seguintes, ou seja, considerando-se um vetor com 6 elementos, compare-se:

O **primeiro** com os **cinco seguintes**;

O **segundo** com os **quatro seguintes**;

O **terceiro** com os **três seguintes** e assim sucessivamente.

Caso um dos elementos seguintes seja menor, troca-se os conteúdos de posição. 



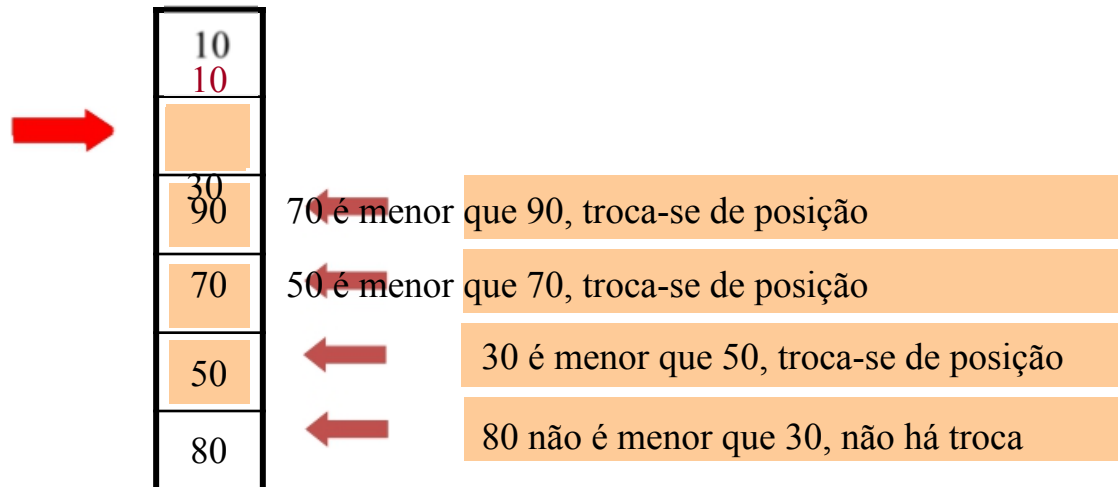
A primeira sequência de comparações não garante a classificação, apenas faz com que o menor dos números esteja na primeira posição, sendo necessário que haja sucessivas sequências de comparações.

# Classificação de Array Unidimensional

## Explicando a Técnica

A **primeira** sequência de comparação fez com que o **primeiro** menor número esteja na **primeira** posição.

A **segunda** sequência de comparação fará com que o **segundo** menor número esteja na **segunda** posição.

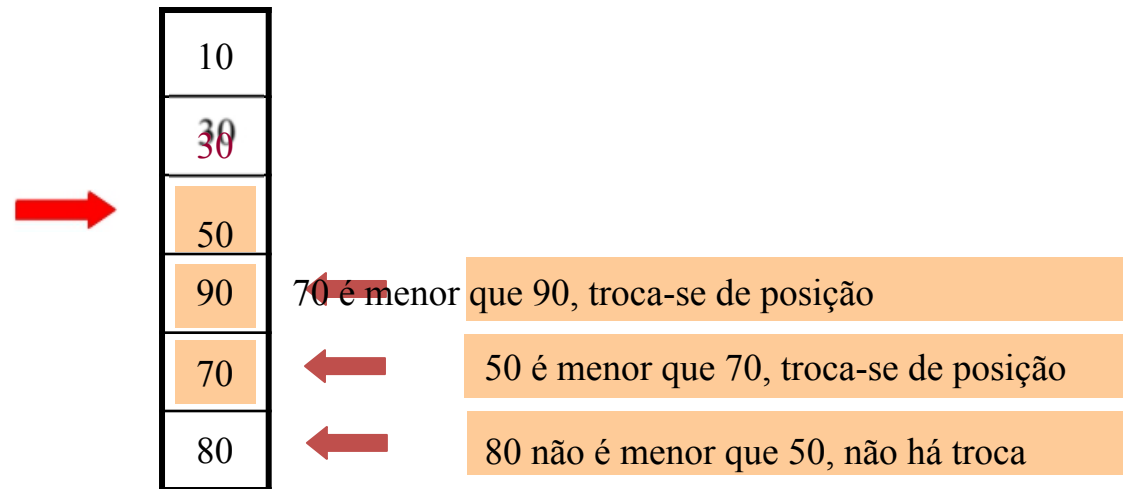


# Classificação de Array Unidimensional

## Explicando a Técnica

A **segunda** sequência de comparação fez com que o **segundo** menor número esteja na **segunda** posição.

A **terceira** sequência de comparação fará com que o **terceiro** menor número esteja na **terceira** posição e etc.



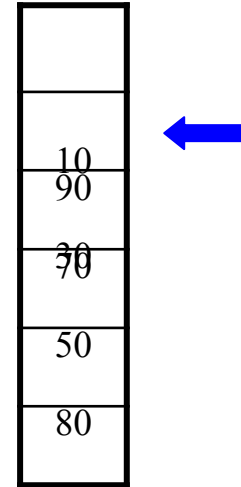
Continuando assim, até que o array esteja classificado.

# Classificação de Array Unidimensional

## Montando as Estruturas de Repetição

Note que necessitamos de dois apontadores para os índices.

```
for (int x = 0; x < 5; x++)  
{  
    for (int y = x + 1; y < 6;  
y++)  
    {  
  
    }  
}
```



Como o sexto elemento não precisa ser comparado com nenhum outro, o primeiro laço só necessita ir até 5, e enquanto o segundo laço não necessita começar do primeiro elemento, deve começar sempre após o primeiro laço, isto é,  $X+1$ .

Observe ainda que os laços estão encadeados.

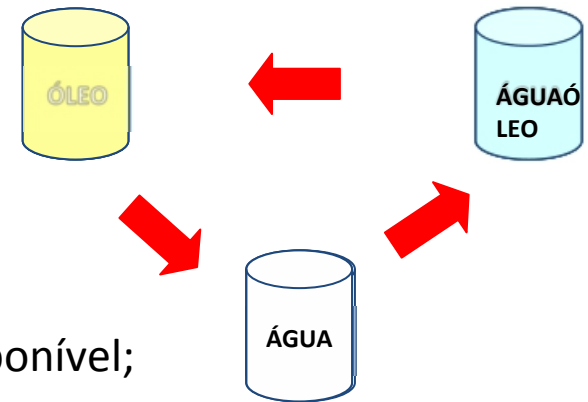
# Classificação de Array Unidimensional

## Fazendo a Troca de Posições

Independente de ser um vetor, necessitamos de uma terceira variável para auxiliar na troca de conteúdo entre duas variáveis.

Fazendo uma analogia, onde as duas variáveis serão representadas por dois copos. Um contendo água e o outro contendo óleo.

Para fazermos a troca de conteúdo entre OS ÁGUAÓLEO Copo, necessitamos de um terceiro copo.



Passamos a água para o copo vazio, deixando-o disponível;  
Passamos o óleo para o novo copo vazio, deixando-o disponível;  
E finalmente, passamos a água para o copo atualmente disponível, que continha óleo.

Codificando temos:

```
vazio= agua;  
agua = oleo;
```

```
oleo = vazio;
```

# Classificação de Array Unidimensional

## Comparando e Fazendo a Troca de Posições

```
for (int x = 0; x < 5; x++)
{
    for (int y = x + 1; y < 6; y++)
    {
        if (num[y] < num[x])
        {
            aux    = num[y];
            num[y] = num[x];
            num[x] = aux;
        }
    }
}
```

Lembre-se que na demonstração da técnica de classificação, quando os valores trocavam de posições, o valor de uma posição sobrepõe o valor da outra posição