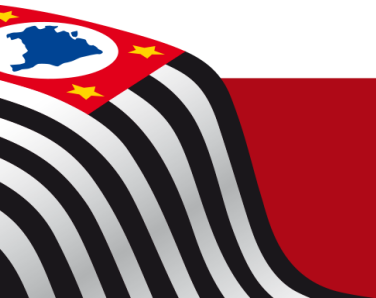


## Curso: ADS

### Estrutura de Dados

### Aula 1 – Vetores em C#

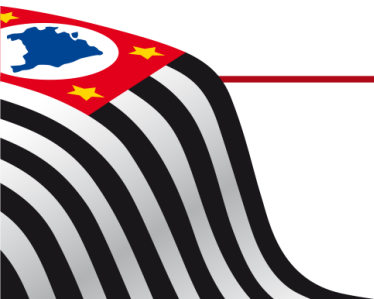
Profº Msc. Anderson L. Coan  
[anderson.coan@fatec.sp.gov.br](mailto:anderson.coan@fatec.sp.gov.br)



# Vetores em C#

- Vetores (ou arrays) são as estruturas de dados mais simples que existem;
- Basicamente são grupos de elementos que podem ser acessados através de seus índices;
- Em C# são declarados da seguinte forma:

```
int[] vetor = new int [8];
```

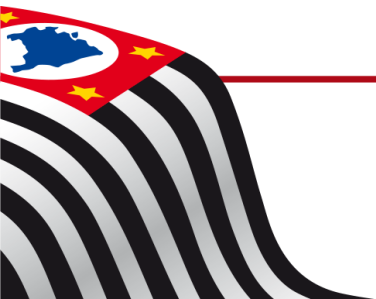


# Vetores em C#

- Dessa forma criamos um vetor que armazenará dados do tipo *int* e com 8 posições, ou seja, capaz de armazenar 8 números, cada um em uma posição diferente, sendo referenciados por seu respectivo índice.
- Para um vetor do tipo string temos:

```
string[] nomes = new string [5];
```

```
nomes[0] = "Juarez";  
nomes[1] = "Christina";  
nomes[2] = "Marcella";  
nomes[3] = "Mirosva";  
nomes[4] = "Estela";
```

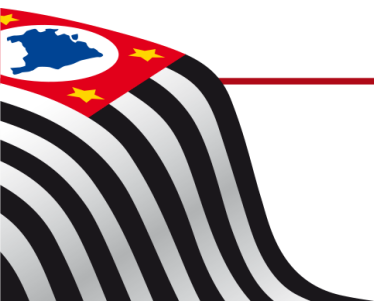


## Vetores em C#

- Para inserirmos um valor no vetor usamos a seguinte sintaxe:

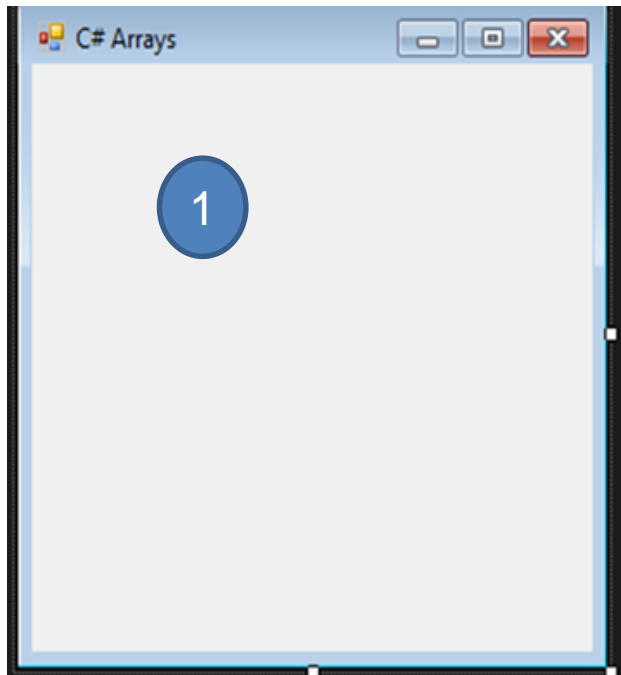
`vetor [0] = 15;`

- Com isso colocamos o número 15 na posição 0 do vetor.
- Importante lembrar que a primeira posição de um vetor sempre tem o índice 0.
- Desta forma, o número 15 está alocado na primeira posição do vetor, sendo seu índice 0.



# Vetores em C#

## Construindo nosso formulário para trabalhar com vetores



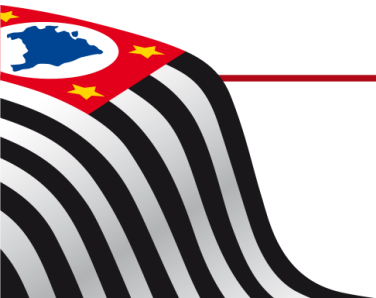
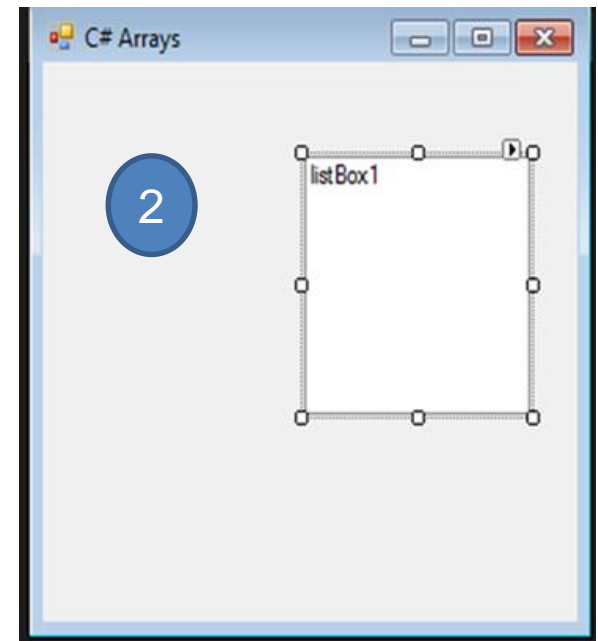
As propriedades do formulário serão:

Text: C# Arrays  
Name: frmArray

Iremos adicionar um controle do tipo *listBox*. Este controle permite exibir uma lista de itens, no caso o nosso vetor.

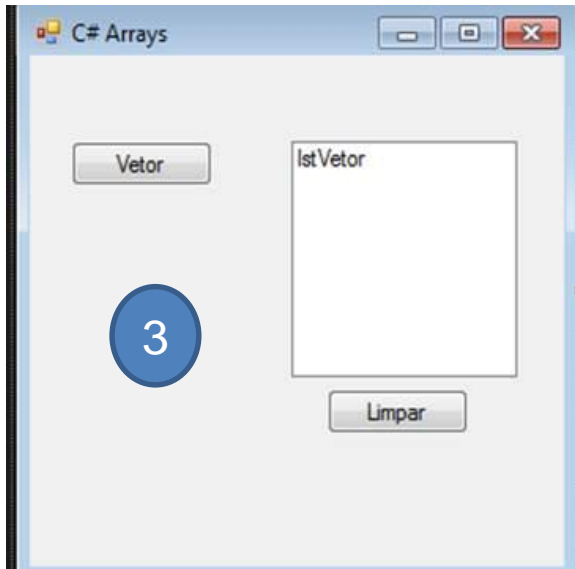
A propriedade do *listBox* será:

Name: lstVetor



# Vetores em C#

## Construindo nosso formulário para trabalhar com vetores



Haverá dois botões. Um para exibir o vetor e outro para limpar os dados.

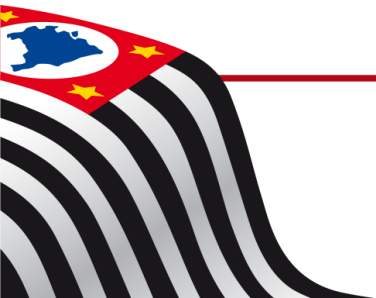
Suas propriedades serão:

Text: Vetor

Name: btnVetor

Text: Limpar

Name : btnLimpar



# Vetores em C#

## Codificando (criando o nosso vetor)

Dê um duplo-clique no botão Vetor. Após isso crie um vetor com 6 posições como segue:

```
18  
19  
20 private void btnVetor_Click(object sender, EventArgs e)  
21 {  
22     int[] numeros = { 7, 9, 10, 8, 6, 5 };  
23 }  
24  
25  
26
```

## “Alimentando” a listBox

Iremos alimentar a *listBox* com o vetor criado. Para isso iremos utilizar a propriedade *Items* em conjunto com o método *Add*. Ainda no código do botão Vetor, abaixo da declaração do vetor iremos adicionar os itens ao *listBox*:

```
20 private void btnVetor_Click(object sender, EventArgs e)  
21 {  
22     int[] numeros = { 10, 9, 8, 7, 6, 5 };  
23     lstVetor.Items.Add("Primeiro elemento: " + numeros[0]);  
24     lstVetor.Items.Add("Segundo elemento: " + numeros[1]);  
25     lstVetor.Items.Add("Terceiro elemento: " + numeros[2]);  
26     lstVetor.Items.Add("Quarto elemento: " + numeros[3]);  
27     lstVetor.Items.Add("Quinto elemento: " + numeros[4]);  
28     lstVetor.Items.Add("Sexto elemento: " + numeros[5]);  
29 }  
30  
31
```

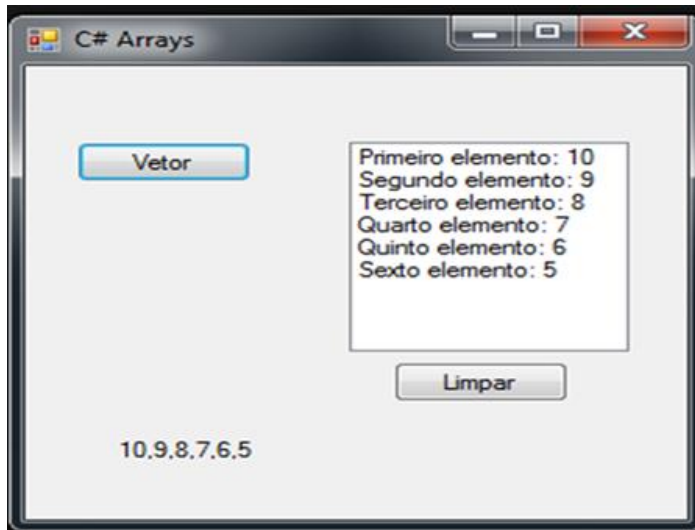
# Vetores em C#

## Botão Limpar

Em modo Design, dê um duplo clique sobre o botão *Limpar* e insira o código abaixo:

```
31  
32  
33     private void btnLimpar_Click(object sender, EventArgs e)  
34     {  
35         lstVetor.Items.Clear();  
36         lblExibe.Text = "";  
37     }  
38  
39
```

A partir deste ponto, seu programa já é funcional. Execute o mesmo, testando o funcionamento dos botões.



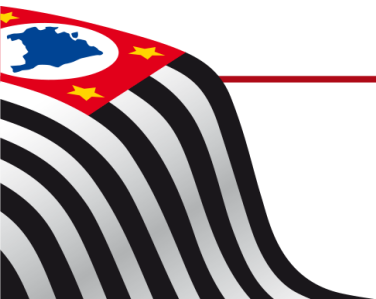


# DÚVIDAS



# Exercícios

1. Faça um programa que leia um vetor com quinze posições para números inteiros. Depois da leitura, divida todos os seus elementos pelo maior valor do vetor. Mostre o vetor após os cálculos.
2. Faça um programa que receba 10 números inteiros digitados pelo usuário. Crie um segundo vetor que irá calcular o dobro do valor de cada posição digitada. Em seguida, exiba os 2 vetores.



## Curso: ADS

### Estrutura de Dados

### Aula 1 – Vetores em C#

Profº Msc. Anderson L. Coan  
[anderson.coan@fatec.sp.gov.br](mailto:anderson.coan@fatec.sp.gov.br)

