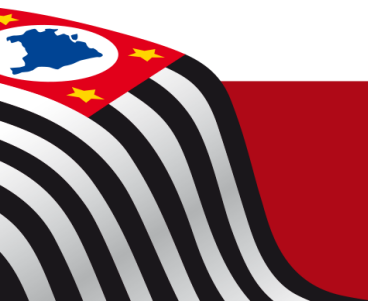


Curso: ADS

Estrutura de Dados

Aula 2 – Ordenando Vetores em C#

Profº Msc. Anderson L. Coan
anderson.coan@fatec.sp.gov.br



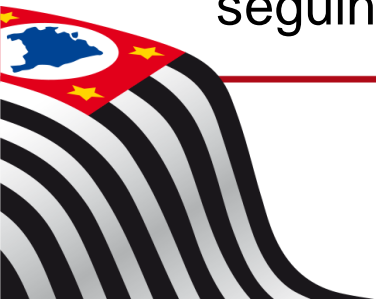
Ordenando Vetores em C#

Por que ordenar um vetor?

- ✓ Às vezes, a necessidade de ordenar informações é inerente a uma aplicação. Por exemplo, para preparar os extratos de clientes, os bancos precisam ordenar os cheques pelo número do cheque;
- ✓ Para emitir uma lista de classificação, uma escola precisa ordenar os candidatos pela nota que cada um tirou no exame;

Ordenando seu vetor com Quicksort (Dividir para Conquistar)

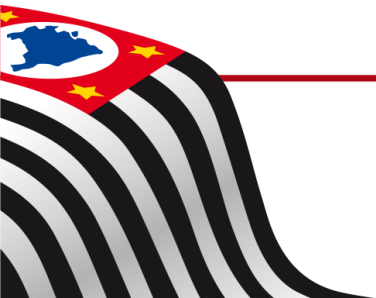
- Um dos algoritmos clássicos para esta tarefa é o Quicksort (Dividir para Conquistar). Seu funcionamento básico parte das seguintes premissas:



Ordenando Vetores em C#

- Um vetor vazio [] e um vetor com apenas um elemento [1] não precisam ser ordenados (esses são os casos-base do algoritmo);
- Um vetor com dois elementos [4,3] também é muito simples de ordenar; basta verificar se o primeiro elemento é menor que o segundo; caso não seja eles são trocados de lugar.
- Então o que este algoritmo faz é sempre quebrar (dividir) um vetor para chegar ao caso base. Observe um vetor com 3 elementos:

V=[33,15,10]



Ordenando Vetores em C#

O algoritmo escolhe um dos elementos do vetor como pivô. Em seguida particiona o vetor em:

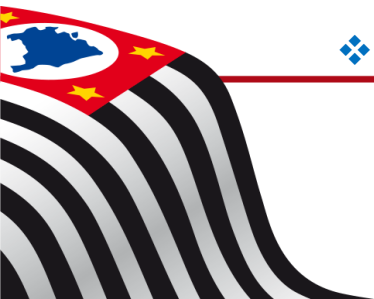
- ❖ Um sub-vetor contendo todos os números menores que o pivô;
- ❖ O pivô;
- ❖ Um sub-vetor contendo todos os números maiores que o pivô;

Suponha 33 como pivô, então teríamos:

- ❖ Um sub-vetor contendo todos os números menores que o pivô; [15,10]

(aqui basta verificar se o primeiro é menor que o segundo, caso não trocar, então temos: [10,15])

- ❖ O pivô; [33]



Ordenando Vetores em C#

(o caso-base nos diz que um vetor com único elemento não precisa ser ordenado.)

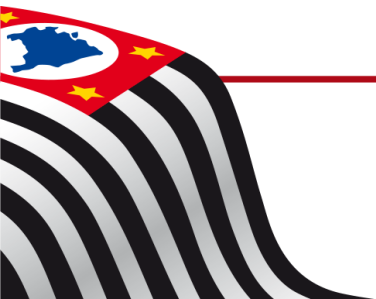
- ❖ Um sub-vetor contendo todos os números maiores que o pivô; [] – que para este caso específico estaria vazio.

(o caso-base nos diz que um vetor vazio não precisa ser ordenado.)

[15,10]	[33]	[0]
Sub-vetor menor	pivô	Sub-vetor maior (vazio)

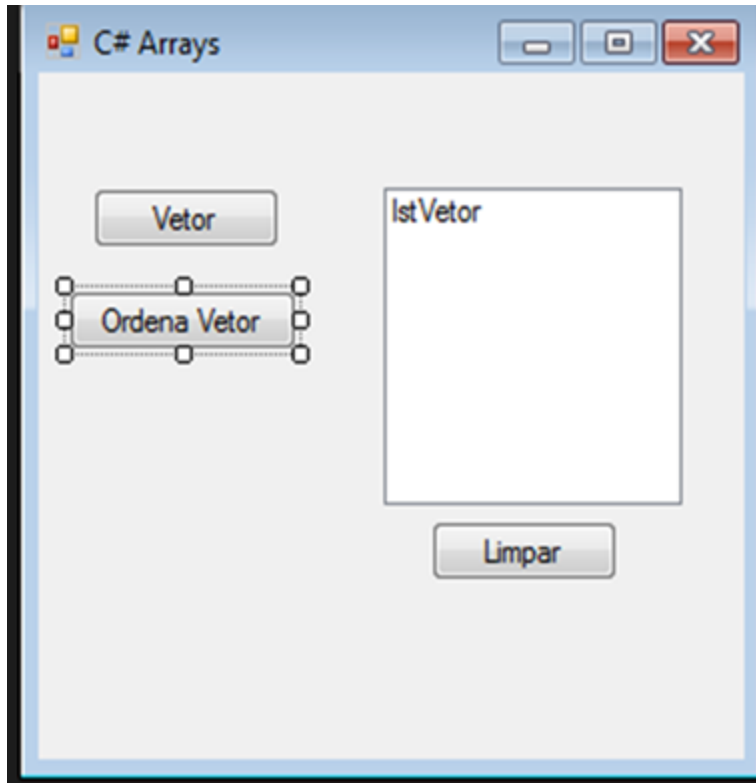
Desta forma o Quicksort (dividir para conquistar) retornaria:

V = [10,15,33] --> Vetor Ordenado



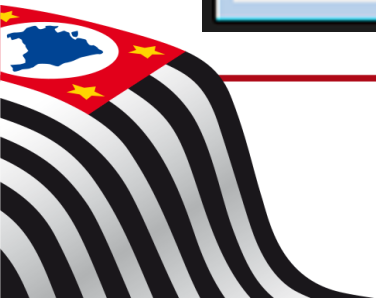
Ordenando Vetores em C#

Aproveitamos então o mesmo formulário quando criamos vetores para adicionar um outro botão para Ordenar nosso vetor:



Suas propriedades serão:

- Text: OrdenaVetor
- Name: btnOrdena



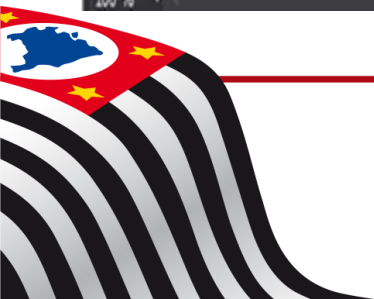
Ordenando Vetores em C#

Codificando a ordenação do seu vetor

Dê um duplo-clique no botão *Ordena Vetor*. Após isso digite o seguinte código:

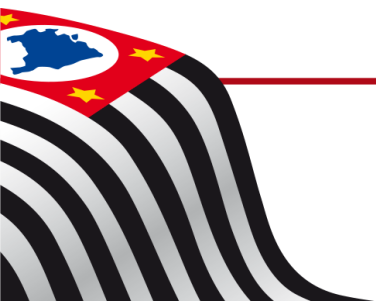
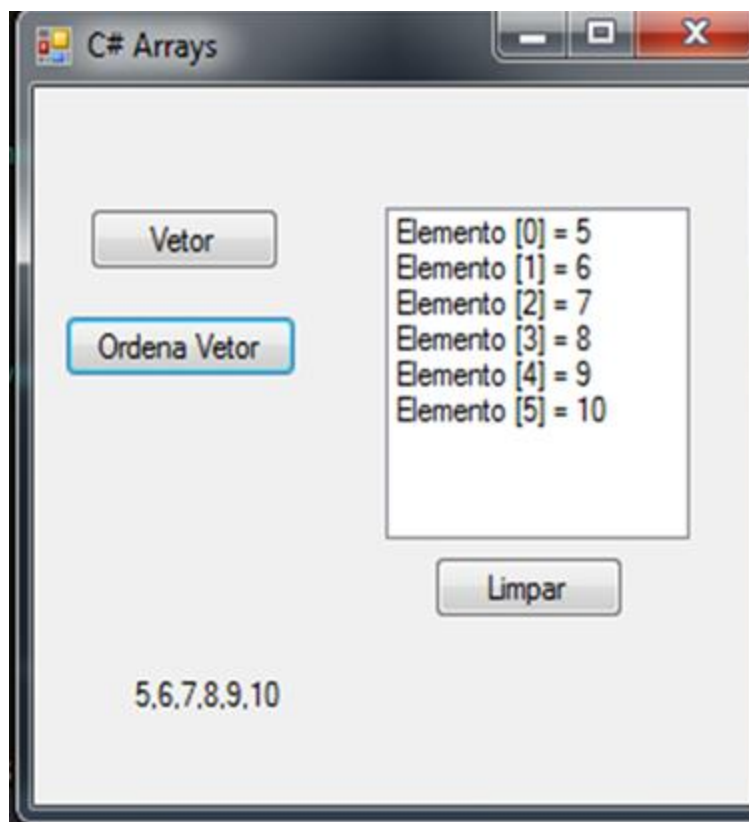
```
54 }
55
56 private void btnOrdena_Click(object sender, EventArgs e)
57 {
58     int[] numeros = { 10, 9, 8, 7, 6, 5 };
59     Array.Sort(numeros);
60
61     for (int i = 0; i < 6; i++)
62     {
63         lst1.Items.Add("Elemento [" + i + "] = " + numeros[i]);
64     }
65     lblExibe.Text = string.Join(", ", numeros);
66 }
67
68
69 }
```

A classe interna do C# “*Array*” possui dentre suas propriedades, o “*Sort*”, que realiza ordenação de vetores.



Ordenando Vetores em C#

Vetor Ordenado



DÚVIDAS

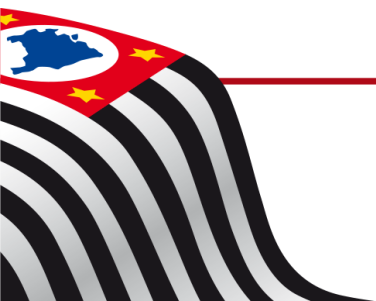


Exercícios

1. Utilizando o trecho de código abaixo, que trata da função do bubble Sort utilizado em C, implemente este método em C# para realizar a ordenação de um vetor de 15 números inteiros (não utilizar Array.Sort).

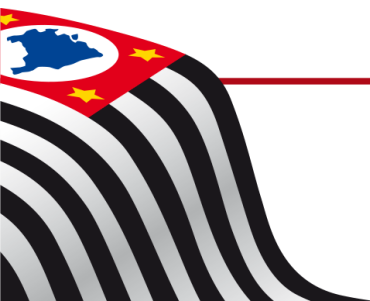
```
void bsort(int vet[], int t)
{
    int i,j,k=0;

    for (i=0;i<t-1;i++)
    {
        for (j=0;j<t-(i+1);j++)
        {
            if (vet[j] > vet[j+1])
            {
                k=vet[j];
                vet[j]=vet[j+1];
                vet[j+1]=k;
            }
        }
    }
}
```



Exercícios

2. Faça um programa que insira 5 nomes em um vetor. Depois da inserção, realize a ordenação em C#.
3. Elabore um programa que preencha um vetor com 18 posições:
 - a) Mostre o maior elemento do vetor e sua respectiva posição (antes da ordenação);
 - b) o menor elemento do vetor e sua respectiva posição (antes da ordenação);
 - c) Ordene e mostre o vetor ordenado.



Curso: ADS

Estrutura de Dados

Aula 2 – Ordenando Vetores em C#

Profº Msc. Anderson L. Coan
anderson.coan@fatec.sp.gov.br

