

# PANC: Projeto e Análise de Algoritmos

## Aula 10: Algoritmos de Ordenação I - Inserção : *Ordenação Direta, Binária e Shell Sort*

### Lista de Exercício – Listex 09

Breno Lisi Romano

<http://sites.google.com/site/blromano>

Instituto Federal de São Paulo – IFSP São João da Boa Vista  
Bacharelado em Ciência da Computação – 3º Semestre



INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
SÃO PAULO  
Campus São João da Boa Vista



# Instruções Gerais para a Listex

## ■ Instruções:

- Todos os exercícios desta Listex devem ser realizados **individualmente**
- Estimula-se a **discussão** com outros **colegas** de **turma** para o **desenvolvimento**, mas **difícilmente soluções individualizadas** serão iguais → **Trabalho Idênticos serão penalizados**: Oportunidade de Aprender Errando
- Todos os **exercícios** desta Listex precisam ser **entregues na Plataforma Google Classroom**, na **atividade** criada, seguindo a data e hora definidos como **prazo de entrega**
- Quando os **exercícios envolverem programação**, **compactem o projeto com os arquivos** para solução dos exercícios, diferenciando cada um deles, o seguinte **padrão de nomes**:
  - Modelo: Listex01-Exercício01-NomeSobrenomeAluno.zip
  - Exemplo: Listex01-Exercício01-BrenoRomano.zip
- Quando os **exercícios envolverem pesquisar, textos escritos, manipulações matemáticas ou outros casos semelhantes**, **entreguem** o exercício em um arquivo na **extensão .PDF**, seguindo o **padrão de nomes**:
  - Modelo: Listex01-Exercício01-NomeSobrenomeAluno.pdf
  - Exemplo: Listex01-Exercício01-BrenoRomano.pdf



# Trabalhos para Casa (1)

- Exercício 01** – Realizar a ordenação binária para o seguinte array. Utilize o algoritmo de busca binária abaixo para encontrar as posições corretas em cada passo. Além disso, utilize também a tabela auxiliar em cada passo:

	S1		S2			
i	0	1	2	3	4	5
Vet[i]	60	30	40	50	90	80
	↑ L	↑ R				

## //Busca Binária

```
L = 0;  
R = i;  
while (L < R)  
{  
    m = piso(L + R)/2;  
    if (x >= Vet[m]) L = m+1;  
    else R = m;  
}
```

## Exemplo – Início da Tabela para o Passo 1

L	R	m
0	1	0
x deve ser inserido na posição de R ou L (iguais).		



## Trabalhos para Casa (2)

- **Exercício 02 – Simular o Array  $A[] = \{44, 12, 55, 42, 94, 18, 02, 68\}$  para os algoritmos de ordenação a seguir, mostrando o passo a passo das ordenações:**
  - a. Algoritmo de Insertion Sort
  - b. Algoritmo da Ordenação Binária
  - c. Algoritmo do Shell Sort ( $np = 2$ )



## Trabalhos para Casa (3)

- **Exercício 03** – Implementar, na Linguagem C, os seguintes algoritmos, em programas distintos, onde deve-se solicitar o tamanho do array e o array a ser ordenado. O resultado deve ser a impressão, passo a passo da ordenação, com o resultado final do array ordenado:
  - a. Algoritmo de Insertion Sort
  - b. Algoritmo da Ordenação Binária
  - c. Algoritmo do Shell Sort

Exemplo para o Insertion Sort:

```
C:\Users\blromano\Desktop\Aula09-10-11-Ordenacoes\bin\Debug\Aula09-10-11-Ordenacoes.exe
----- Metodos de Ordenacao por Insercao -----
1-) Ordenacao Direta
Vetor Desord.[] = 42 48 7 90 37 25 9 21 71 61 13 50 98 33 7 - 48
Vetor Desord.[] = 42 48 7 90 37 25 9 21 71 61 13 50 98 33 7 - 7
Vetor Desord.[] = 7 42 48 90 37 25 9 21 71 61 13 50 98 33 7 - 90
Vetor Desord.[] = 7 42 48 90 37 25 9 21 71 61 13 50 98 33 7 - 37
Vetor Desord.[] = 7 37 42 48 90 25 9 21 71 61 13 50 98 33 7 - 25
Vetor Desord.[] = 7 25 37 42 48 90 9 21 71 61 13 50 98 33 7 - 9
Vetor Desord.[] = 7 9 25 37 42 48 90 21 71 61 13 50 98 33 7 - 21
Vetor Desord.[] = 7 9 21 25 37 42 48 90 71 61 13 50 98 33 7 - 71
Vetor Desord.[] = 7 9 21 25 37 42 48 71 90 61 13 50 98 33 7 - 61
Vetor Desord.[] = 7 9 21 25 37 42 48 61 71 90 13 50 98 33 7 - 13
Vetor Desord.[] = 7 9 13 21 25 37 42 48 61 71 90 50 98 33 7 - 50
Vetor Desord.[] = 7 9 13 21 25 37 42 48 50 61 71 90 98 33 7 - 98
Vetor Desord.[] = 7 9 13 21 25 37 42 48 50 61 71 90 98 33 7 - 33
Vetor Desord.[] = 7 9 13 21 25 33 37 42 48 50 61 71 90 98 7 - 7
Vetor Desord.[] = 7 7 9 13 21 25 33 37 42 48 50 61 71 90 98
Vetor Ord. [] = 7 7 9 13 21 25 33 37 42 48 50 61 71 90 98
```

## Slide 5

---

**b1**

blromano; 01/09/2020