

# **ALGORITMOS E LÓGICA DE PROGRAMAÇÃO**

## **Aula 5**



# Variáveis compostas

**Aprendemos no início do curso que as variáveis, são espaços reservados na memória do computador que podemos armazenar dados e que esses dados armazenados tem um tipo específico.**

**Inteiro = INT**

**Real = Float**

**Lógicos = Boolean**

**Texto = String**

**caracteres = char**

# Variáveis compostas

**Imagine que precisamos armazenar o nome de todos os alunos do CDD4.0. Quantas variáveis teríamos que criar?**

**considerando que temos inicialmente 25 alunos por turma e que no total temos 4 turmas.**

**Teríamos um total de 100 variáveis.**

# Variáveis compostas

```
int[] novo_array = new int[100];
```

José carlos	Ana Corolina	Helan	Clara	João Pedro	Daniel Martins	Mateus silva	Douglas oliveira	Claudio jose	Mario silva
----------------	-----------------	-------	-------	---------------	-------------------	-----------------	---------------------	-----------------	----------------

***Por definição um array, ou vetor só pode conter um tipo de dado, ou seja, se você criar um array do tipo INT, só pode colocar dados numéricos inteiros, dentro dele.***

# Variáveis compostas em Python

*Python, trata os vetores e matrizes como uma lista, e a grande vantagem é que uma lista aceita diversos tipos de dados.*

NOME	NOTA <sub>1</sub>	NOTA <sub>2</sub>	NOTA <sub>3</sub>	MEDIA
<b>José carlos</b>	<b>8.5</b>	<b>7.5</b>	<b>10</b>	<b>8.67</b>

## **ACESSANDO OS DADOS DE UMA LISTA**

**Aqui vamos começar com um exemplo bem simples de lista, onde estamos atribuindo 3 itens a variável lista\_compras.**

```
arrayCompras = ['banana','laranja','maçã']  
for i in arrayCompras:  
    print(i)
```



```
lista_compras = ["", "", ""]
```

```
lista_compras = []
```

```
lista_compras = ['feijao', 'arroz', 'farinha']
```

# Métodos do Python para utilizar nas listas

**Para adicionar um item à lista:**

**.append():** adiciona o item ao final da lista;

**.insert():** insere um item na lista na posição indicada

**del:** remove um item da lista baseado na posição indicada;

**.remove():** remove um item baseado no seu valor e não na sua posição;

**.pop():** remove da lista\_compras o último item, mas não o exclui.



# Métodos do Python para utilizar nas listas

**Para adicionar um item a lista:**

**.append():** adiciona o item ao final da lista;

Exemplo::

```
lista_compras.append('carro')
```

# Métodos do Python para utilizar nas listas

**Para adicionar um item à lista:**

**.insert():** insere um item na lista na posição indicada

```
lista_compras.insert(1,'carro')
```



# **Métodos do Python para utilizar nas listas**

**Para Deletar um item da lista:**

**`del lista_compras[3]`**



**Métodos do Python para utilizar nas listas**  
**Para remover um item da lista:**

**Removendo um item com base no seu valor e não na sua posição, com o .remove() (ATENÇÃO! retira apenas a primeira ocorrência e não todas):**

# **Métodos do Python para utilizar nas listas**

## **Para remover um item da lista:**

**Utilizando .pop(), iremos remover o último item da lista\_compras, mas sem excluí-lo. Nesse caso, o valor carro será armazenado na variável item:**

```
item = lista_compras.pop(-1)  
print(item)
```



## **Exercício 01**

**Criar um array tamanho 5 e preencher com os nomes dos 5 alunos, informados pelo usuário**



## **Exercício 02**

**altere o exercício anterior e mostre na tela,  
ao final, o nome de cada aluno e sua  
respectiva posição no array.**



### **Exercício 03**

**altere o exercício anterior para permitir  
achar o nome de um aluno na lista**



### **Exercício 04**

**Escreva um código que permita a leitura das notas de uma turma de 5 alunos e guarde num vetor, Calcular a média da turma e contar quantos alunos obtiveram nota acima desta média calculada Escrever a média da turma e o resultado da contagem**

### **Exercício 05**

**preencher um vetor A com 10 números. logo em seguida, ler mais um número e guardar em uma variável X.**

**Armazenar em um vetor M o resultado de cada elemento de A multiplicado pelo valor X.**

**no final imprimir o vetor M.**

## **Exercício 06**

**Faça um código para ler 5 números e armazenar em um vetor. Após a leitura total dos 5 números, o código deve escrever esses 5 números lidos na ordem inversa.**

## **Exercício 07**

**Faça um código para ler 5 números e armazenar em um vetor. Após a leitura total dos 5 números, o código deve escrever esses 5 números lidos na ordem inversa.**

## **Exercício 08**

**Faça um código para ler 5 nomes de usuários e suas respectivas senhas, e armazenar cada lista em um array diferente, após completar a digitação, imprimir , nome, senha e posição dos dados no array**



## **Exercício 09**

**Altere o sistema anterior e faça um sistema de login, pedindo a senha do usuário e mostrando seu nome e a mensagem, login efetuado com sucesso.**

### **Exercício 10**

**Faça um código para ler um valor N qualquer (que será o tamanho dos vetores). Após, ler dois vetores A e B (de tamanho N cada um) e depois armazenar em um terceiro vetor Soma a soma dos elementos do vetor A com os do vetor B (respeitando as mesmas posições) e escrever o vetor Soma.**

## **Exercício 11**

**Faça um código para ler 10 números e guardar num vetor. Após isto, ler mais um número qualquer, calcular e escrever quantas vezes esse número aparece no vetor.**



## **Exercício 12**

**Escreva um algoritmo que solicite ao usuário a entrada de 5 nomes, e que exiba a lista desses nomes na tela.**

**Após exibir essa lista, o programa deve mostrar também os nomes na ordem inversa em que o usuário os digitou, um por linha.**

### **Exercício 13**

**Faça um algoritmo que leia 10 valores do tipo inteiro e armazene-os em um vetor.**

**A seguir, o algoritmo deverá informar**

- (1) todos os números pares que existem no vetor;**
- (2) o menor e o maior valor existente no vetor;**
- (3) quantos dos valores do vetor são maiores que a média desses valores:**