

# Lógica de Programação

Aula 5 - 27/03/2023



[https://bit.ly/23\\_1\\_cc\\_logica\\_m](https://bit.ly/23_1_cc_logica_m)



## O que temos para **hoje**?

- Um novo tipo de variável;
- Operadores relacionais;
- Operadores lógicos;
- Estruturas de decisão;



# Um novo tipo de variável



## Um novo **tipo** de variável

- Até agora, vimos 3 tipos de variáveis:
  - Int;
  - Float;
  - String;



## Um novo **tipo** de variável

- Novo tipo de variável: lógica ou booleana;
- Aceita apenas dois valores: True ou False;



## Um novo **tipo** de variável

- Exercício: criar uma variável e atribuir o valor True para ela. Depois, escrever essa variável na tela;



# Operadores relacionais

---



## Operadores relacionais

- Operador Relacional é todo operador que obtém a relação em expressões condicionais;
- Expressão condicional:

[membro da esquerda] **OPERADOR RELACIONAL** [membro da direita];

- O resultado de uma expressão condicional é sempre True ou False, ou seja, é um valor lógico ou booleano;





## Operadores relacionais

>	Maior que
<	Menor que
>=	Maior ou igual a
<=	Menor ou igual a
==	Igual a
!=	Diferente de



## Operadores relacionais

### Exemplos:

- $2 > 5$
- $\text{salario} < 5000$
- $\text{idade} \geq 18$
- $\text{nome} \neq \text{"João"}$
- $\text{nota} == 10$



## Operadores relacionais

- IMPORTANTE: não confunda = com ==;
- Um sinal de igual (=): operador de atribuição, usado para atribuir um valor a uma variável;
- Dois sinais de igual (==): operador relacional, usado para a igualdade entre dois valores;



## Operadores relacionais

- Exercício: crie um programa que tenha duas variáveis e atribua valores para elas;
- Faça a comparação entre essas variáveis com todos os operadores relacionais e escreva o resultado;



# Operadores lógicos



## Operadores lógicos

- Os operadores lógicos unem expressões condicionais formando assim, uma nova expressão que é composta por 2 ou mais sub-expressões;
- O resultado lógico de expressões compostas será a relação entre as sub-expressões



## Operadores lógicos

- Quando estudamos os Operadores Relacionais aprendemos a obter o valor lógico entre 2 operandos, no caso, entre o operando que está a esquerda do operador e o operando que está a direita do Operador Relacional;
- Como resposta, obtemos valores do tipo Booleano, isto é, verdadeiro [True] ou falso [False].



## Operadores lógicos

- Os operadores lógicos por sua vez, permitem-nos unir 2 expressões ligando-as com os conectivos lógicos matemáticos que são, o conectivo E (and), o conectivo OU (or) e o conectivo Não (not).





## Operadores lógicos

☉ Exemplo:

```
1  nome = "Fabrício"
2  idade = 37
3  print(nome != "Pedro" and idade < 40)
```



## Operadores lógicos

### ● Tabela verdade: and

exp_cond1	exp_cond2	exp_cond1 <b>and</b> exp_cond2
True	True	True
True	False	False
False	True	False
False	False	False



## Operadores lógicos

### ● Tabela verdade: or

exp_cond1	exp_cond2	exp_cond1 <b>or</b> exp_cond2
True	True	True
True	False	True
False	True	True
False	False	False



## Operadores lógicos

### ● Tabela verdade: not

exp_cond1	<b>not</b> exp_cond1
True	False
False	True



## Operadores lógicos

Operador	Significado	Precedência
()	Grupos entre parênteses	+
**	Potenciação	
-	Negação	
* / % //	Multiplicação, divisão real, resto, divisão inteira	
+ -	Subtração, adição	
> >= < <= == !=	Comparações	
not	NÃO lógico	
and	E lógico	
or	OU lógico	
=	Atribuição	-



## Operadores lógicos

- Exercícios: qual o valor das seguintes expressões, se  $x = 15$  e  $y = 10$ :
- $x == 15$  or  $y != 10$
  - $x >= 15$  and not  $y != 10$
  - $x != 10$  and  $y != 15$  or  $x >= 15$  or  $y < 20$
  - $x > 20$  and  $y == 10$  or  $x != 15$  and not  $x > 10$



## Operadores lógicos

- Exercícios: qual o valor das seguintes expressões:
  - $\text{False or } (10 \% 5 * 2 \neq 5 * 2 + 1)$
  - $\text{not False and } (3 * 3 / 3 < 15 - 5 \% 7)$
  - $((34 < 9) \text{ and } (5 + 29 == 34)) \text{ or } ((5 == 15/3) \text{ and } (8 > 12))$



# Estruturas de decisão

---





## Estruturas **decisão**

- Vimos que um programa é sempre escrito descrevendo as ações linha após linha;
- Há momentos em que precisamos tomar uma decisão sobre como o programa deve seguir;



## Estruturas **decisão**

- Exemplo: lista de exercícios 1, questão 14:
- E se o peso que João pescou for menor que 50?

14. João Papo-de-Pescador, comprou um microcomputador para controlar o rendimento diário de seu trabalho. Toda vez que ele traz um peso de peixes maior que o estabelecido pelo regulamento de pesca do estado de São Paulo (50 quilos) deve pagar uma multa de R\$ 4,00 por quilo excedente. João precisa que você faça um programa que leia a variável *peso* (peso de peixes) e calcule o excesso. Gravar na variável *excesso* a quantidade de quilos além do limite e na variável *multa* o valor da multa que João deverá pagar. Imprima os dados do programa com as mensagens adequadas.



## Estruturas **decisão**

- Exemplo: lista de exercícios 1, questão 14:
- E se o peso que João pescou for menor que 50?

```
1  peso = input("Seu João, digite o peso dos peixes: ")
2  peso = float(peso)
3  # o valor de peso é menor que 50?
4  peso_menor_50 = peso < 50
5  # e agora?
```



## Estruturas **decisão**

- ◎ Solução: estruturas de decisão:
  - if (se);
  - else (senão);
  - elif (senão se);



## Estruturas **decisão**

### ☉ Sintaxe if:

```
if condição:  
    # código que será executado caso a condição seja verdadeira
```



## Estruturas decisão

```
1  peso = input("Seu João, digite o peso dos peixes: ")
2  peso = float(peso)
3  # o valor de peso é menor que 50?
4  ✓ if peso < 50:
5      print("Você não precisa pagar multa!")
```



## Estruturas **decisão**

- E se a expressão do if não for verdadeira?
- Ou seja, o que acontece em todos os outros casos?



## Estruturas **decisão**

### ☉ Sintaxe else:

```
if condição:  
    # código que será executado caso a condição seja verdadeira  
else:  
    # código que será executado caso a condição NÃO seja verdadeira
```





## Estruturas **decisão**

- Importante: else só acontece junto com if;
- O conjunto das estruturas if e else define o tratamento de uma condição e o restante dos casos;



## Estruturas decisão

```
1  peso = input("Seu João, digite o peso dos peixes: ")
2  peso = float(peso)
3  # o valor de peso é menor que 50?
4  if peso < 50:
5      print("Você não precisa pagar multa!")
6  else:
7      excesso = peso - 50
8      multa = excesso * 4
9
10     peso = str(peso)
11     excesso = str(excesso)
12     multa = str(multa)
13
14     print("O peso de peixes é: "+peso)
15     print("O excesso foi: "+excesso)
16     print("A multa é: R$"+multa)
```



## Estruturas **decisão**

- Exercício: faça um programa que peça dois números e imprima o maior deles.



## Estruturas **decisão**

- Exercício: faça um programa que peça um valor e mostre na tela se o valor é positivo ou negativo.



## Estruturas **decisão**

- ◎ E se quiséssemos testar mais de uma condição?
- ◎ Exemplos:
  - Slide 35: e se os números digitados forem iguais?
  - Slide 36: e se o valor digitado for 0?



## Estruturas **decisão**

### ☉ Sintaxe elif:

```
if condição:  
    # código que será executado caso a condição seja verdadeira  
elif outra condição:  
    # código que será executado caso a outra condição seja verdadeira  
else:  
    # código que será executado caso a condição NÃO seja verdadeira
```



## Estruturas decisão

```
1  n1 = input("Digite um número: ")
2  n1 = float(n1)
3
4  n2 = input("Digite outro número: ")
5  n2 = float(n2)
6
7  √ if n1 > n2:
8      | print(str(n1)+" é maior que "+str(n2))
9  || √ elif n1 == n2:
10     | print(str(n1)+" é igual "+str(n2))
11  √ else:
12     | print(str(n2)+" é maior que "+str(n1))
```



## Estruturas decisão

```
1  n1 = input("Digite um número: ")
2  n1 = float(n1)
3
4  √ if n1 > 0:
5      print(str(n1)+" é positivo")
6  √ elif n1 == 0:
7      print(str(n1)+" não é positivo nem negativo")
8  √ else:
9      print(str(n1)+" é negativo")
```





## Estruturas **decisão**

- Exemplo: faça um programa que peça uma letra para o usuário. Se a letra digitada for "F", escreva "Feminino", se a letra digitada for "M", escreva "Masculino". Senão for nenhum desses dois casos, escreva "Opção não cadastrada".



## Estruturas **decisão**

### ● Regras gerais:

- Uma estrutura de decisão SEMPRE começa com if;
- A estrutura pode ter elif, quando queremos testar mais de uma decisão, mas não é obrigatória;
- A estrutura pode ter else, quando queremos definir o resultado para todos os casos restantes, mas não é obrigatória;



## Estruturas **decisão**

- Exercício: faça um programa que peça a idade do usuário. Caso a idade seja maior ou igual a 18 anos, escreva “Você já pode tirar a carteira de motorista”. Se a idade não for maior ou igual a 18, escreva “Você ainda não pode tirar a carteira de motorista”



## Estruturas **decisão**

- Exercício: faça um programa que peça a média das notas do semestre do usuário. Caso a media seja menor que 4, escreva "Você está reprovado". Se a idade for maior ou igual a 4 e menor que 7, escreva "Você deve fazer a prova final". Se a media for maior ou igual a 7, escreva "Você foi aprovado direto";



## Estruturas **decisão**

- Desafio: desenvolva um programa que receba um número e diga se esse número é par ou ímpar;



## Estruturas **decisão**

- Lista de exercícios 2;



# Obrigado!

*Alguma pergunta?*

Contato:

☎ 040601692@prof.unama.br