

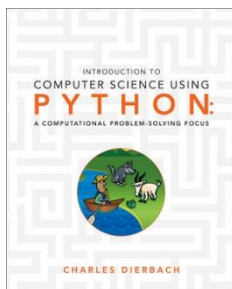
**FACULDADE DE COMPUTAÇÃO E INFORMÁTICA**  
**CIÊNCIA DA COMPUTAÇÃO E SISTEMAS DE INFORMAÇÃO**  
**Algoritmos e Programação I**

**AULAS 04 e 05: OPERADORES E EXPRESSÕES LÓGICAS, ESTRUTURAS CONDICIONAIS SIMPLES E COMPOSTA**



Nossos **objetivos** nesta aula são:

- Saber o que é uma estrutura de controle de fluxo
- Conhecer os operadores relacionais e os operadores lógicos.
- Desenvolver e avaliar o resultado de expressões lógicas envolvendo operadores relacionais e lógicos
- Construir algoritmos com desvio de fluxo.
- Utilizar os principais comandos condicionais: simples e composto.
- Ser capaz de ler, escrever e implementar programas que empregam estruturas condicionais.

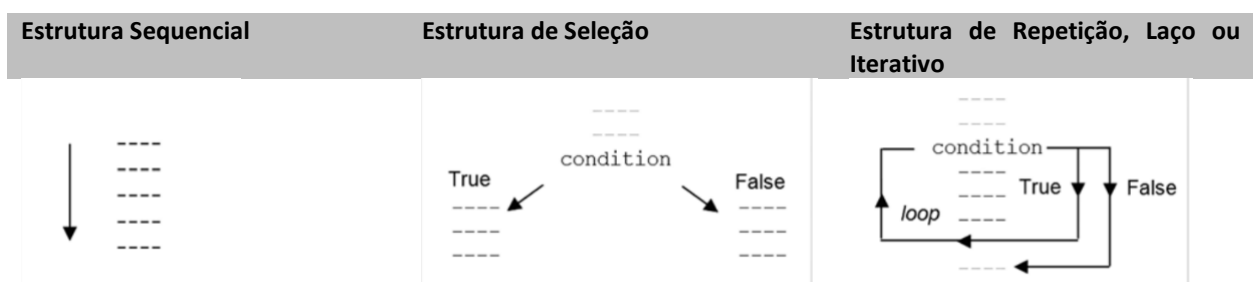


A referência para esta aula são as seções **3.1 (O que é uma estrutura de controle)**, **3.2 (Expressões lógicas)** e **3.3 (Controle de Seleção)** do **Capítulo 3 (Control Structures)** do livro:

DIERBACH, C. *Introduction to Computer Science Using Python: A Computational Problem Solving Focus*. 1st Edition, New York: Wiley, 2012.

**O QUE É UMA ESTRUTURA DE CONTROLE?**

- A ordem em que as instruções de um programa são executadas é chamada de **controle de fluxo**.
- Existem três formas fundamentais de controlar o fluxo de um programa: estrutura de controle sequencial, estrutura de controle de seleção ou condicional e estrutura de controle de repetição, iterativo ou laço.



- A **estrutura de controle sequencial**, caracteriza pelos problemas visto até agora, é uma forma implícita de controle, em que as instruções são executadas na ordem em que são escritas.
- Conhecemos o segundo tipo de controle de fluxo, a estrutura de seleção ou condicional. Porém, antes vamos saber o que é uma expressão lógica.

## EXPRESSÕES LÓGICAS

---

- Nesta aula, vamos estudar dois operadores que são usados em **expressões lógicas** (booleanas): **operadores relacionais** e **operadores lógicos**.
- Uma **expressão lógica** é uma expressão que resulta em um valor lógico **True** (verdadeiro) ou **False** (falso).

## Operadores Relacionais

---

- Os **operadores relacionais** são usados em comparações entre valores. O resultado da comparação será um valor lógico. São eles:

Operadores		Exemplo	Resultado
==	igual a	10 == 10	True
!=	diferente	10 != 10	False
<	menor que	10 < 20	True
>	maior que	"Alan" > "Brenda"	False
<=	menor ou igual a	10 <= 10	True
>=	maior ou igual a	"A" >= "D"	False

## EXERCÍCIO TUTORADO 1

---

O que será exibido ao digitarmos as seguintes linhas de comando?

>>> 10 == 20	>>> '2' < '9'	>>> 'Hello' == "Hello"
>>> 10 != 20	>>> '12' < '9'	>>> 'Hello' < 'Zebra'
>>> 10 <= 20	>>> '12' > '9'	>>> 'hello' < 'ZEBRA'

## EXERCÍCIO COM DISCUSSÃO EM DUPLAS 1

Considerando  $a = 4$ ,  $b = 10$ ,  $c = 50$ ,  $d = 1$  e  $f = 5$ , qual será o resultado lógico (True/False) das seguintes expressões:

Expressão	Resultado
$a == c$	
$a < b$	
$d > b$	
$c != f$	
$a == b$	
$c < d$	
$b > a$	
$c >= f$	
$c <= c$	
$c <= f$	

## Operadores Lógicos

- Os **operadores lógicos** podem ser usados para construir expressões lógicas mais complexas combinando comparações. São eles: and (e), or (ou), not (não).
- O operador lógico **and** resulta verdadeiro somente quando seus operandos forem verdadeiros.
- O operador lógico **or** resulta verdadeiro quando, pelo menos, um dos seus operandos for verdadeiro.
- O operador lógico **not** inverte o valor do operando.
- Tabela verdade dos operadores lógicos:

x	y		x and y	x or y	not x
False	False		False	False	True
True	False		False	True	False
False	True		False	True	
True	True		True	True	

- É preciso ser cauteloso ao usar operadores lógicos. Por exemplo, na matemática, para indicar que um valor está dentro de um determinado intervalo escrevemos:

$$1 \leq \text{num} \leq 10$$

- No entanto, na maioria das linguagens de programação esta expressão não faz sentido. Para entender porque, vamos assumir que num tem o valor 15.

$$1 \leq \text{num} \leq 10 \rightarrow 1 \leq 15 \leq 10 \rightarrow \text{True} \leq 10 \rightarrow ???$$

- Não faz sentido verificar se **True** é menor ou igual a 10. A maneira correta de escrever a expressão deve usar o operador lógico **and**

```
1 <= num and num <= 10
    ou
num >= 1 and num <= 10
```

## EXERCÍCIO TUTORADO 2

O que será exibido ao digitarmos as seguintes linhas de comando?

>>> True and False	>>> (10 < 0) and (10 > 2)
>>> True or False	>>> (10 < 0) or (10 > 2)
>>> not(True) and False	>>> not(10 < 0) or (10 > 2)
>>> not(True and False)	>>> not(10 < 0 or 10 > 2)

## EXERCÍCIO COM DISCUSSÃO EM DUPLAS 2

Qual será o valor das seguintes expressões lógicas:

- a) ((5 > 3) or (4 < 2)) and (1 < 7)
- b) (not(5 > 3) or (1 < 2)) and (not(1<7))

## Precedência dos Operadores

- A precedência dos operadores relacionais e lógicos, bem a associatividade dos operadores é dada conforme segue a tabela:

Operador	Associatividade
<, >, <=, >=, !=, ==	Esquerda para direita
not	Esquerda para direita
and	Esquerda para direita
or	Esquerda para direita

- Uma vez que expressões lógicas, pode conter também operadores aritméticos, vistos na aula anterior, podemos estender para a tabela seguinte:

Operador	Associatividade
<b>**</b>	Direita para esquerda
<b>- (negação)</b>	Esquerda para direita
<b>* / // e %</b>	Esquerda para direita
<b>+ e - (subtração)</b>	Esquerda para direita
<b>&lt;, &gt;, &lt;=, &gt;=, !=, ==</b>	Esquerda para direita
<b>not</b>	Esquerda para direita
<b>and</b>	Esquerda para direita
<b>or</b>	Esquerda para direita

## ESTRUTURAS DE SELEÇÃO OU CONDICIONAL

---

- A partir de agora vamos estudar as **estruturas condicionais** que permitem que o programa execute diferentes sequências de instruções em diferentes casos, dependendo da avaliação de uma expressão lógica.
- Uma **expressão lógica** é uma expressão cujos operadores são lógicos e/ou relacionais e cujos operandos são relações e/ou variáveis do tipo lógico.
- Em programação, o uso de condições para permitir a escolha de executar ou não um trecho de programa é muito utilizado, principalmente quando precisamos incluir no programa **condições de controle**, para evitar situações não permitidas, que podem resultar em erros. Por exemplo, para evitar divisões por zero.
- A estrutura lógica que permite que o fluxo de execução de um algoritmo possa sofrer desvios é conhecida como **estrutura condicional ou de seleção**.
- Quando temos apenas um bloco especial de comando ou instruções apenas para quando a expressão lógica for verdadeira, esse tipo de estrutura chama-se **estrutura de condicional simples**.

## ESTRUTURA CONDICIONAL SIMPLES

---

- Sintaxe:

```
if condição:
    instrução(ões)_verdadeiro
```

- Exemplo:

```
if nota >= 6.0:
    print("Aluno aprovado")
if nota < 6.0:
    print("Aluno reprovado")
```

- O texto “**Aluno aprovado**” só será exibido se a condição **nota >= 6.0** for **verdadeira**; caso contrário, o controle passa para a próxima instrução que tem outra condição **nota < 6.0** a ser avaliada. Se o resultado for **verdadeiro** então será exibido o texto “**Aluno reprovado**”.
- Uma característica singular da linguagem Python é que a quantidade de recuo (**indentação**) está associada à um bloco de instruções. Esta **indentação** é obrigatória para se definir qual instrução ou instruções devem ser executadas quando o resultado lógico da expressão for verdadeiro.

Indentação válida	Indentação inválida
<pre>if condição:     instrução     instrução     instrução</pre>	<pre>if condição:     instrução     instrução     instrução</pre>

- Analisando o exemplo anterior, podemos perceber que para escrever **apenas uma** das mensagens foram necessárias **duas condições**.
- Existe outra estrutura, chamada de **estrutura condicional composta**, em que temos uma única condição e dois caminhos que podem ser seguidos – o caminho do resultado verdadeiro e o caminho do resultado falso.

## ESTRUTURA CONDICIONAL COMPOSTA

---

- Sintaxe:

```
if condição:
    instrução(ões)_verdadeiro
else:
    instrução(ões)_falso
```

- Exemplo:

```
if nota >= 6.0:
    print("Aluno aprovado")
else:
    print("Aluno reprovado")
```

- Neste caso, será avaliada a condição **nota >= 6.0** e se o resultado for **verdadeiro** então será apresentado o texto “**Aluno aprovado**”; **caso contrário (else)**, será apresentado o texto “**Aluno reprovado**”.
- Como na estrutura condicional simples, *instrução(ões)\_verdadeiro* e *instrução(ões)\_falso* devem ser indentadas adequadamente.

Indentações válidas		Indentações inválidas	
<pre> if condição:     instrução     instrução else:     instrução     instrução </pre>	<pre> if condição:     instrução     instrução else:     instrução     instrução </pre>	<pre> if condição:     instrução     instrução else:     instrução     instrução </pre>	<pre> if condição:     instrução     instrução else:     instrução     instrução </pre>

### EXERCÍCIO TUTORADO 3

---

Escreva a estrutura apropriada para cada um dos seguintes itens:

a) Apresente “dentro do intervalo” se  $1 \leq \text{num} \leq 100$ .

b) Apresente “dentro do intervalo” se  $30 < \text{num} < 70$ , caso contrário, apresente “fora do intervalo”.

### EXERCÍCIOS COM DISCUSSÃO EM DUPLAS 3

---

1) Resolva as expressões lógicas abaixo:

a)  $s = ((\text{not } (1 == 2)) \text{ and } (3 < 4))$

b)  $s = ((1 < 2) \text{ or } (3 > 4))$

c)  $s = (((2 == 2) \text{ and } (3 > 4)) \text{ or } (3 > 4))$

2) Escreva um programa que leia dois números distintos e apresente o quadrado do maior número.

3) Escreva um programa que leia um número inteiro e exiba se ele é um número par ou ímpar.

4) Um comerciante comprou um produto e quer vendê-lo com um lucro de 45% se o valor da compra for menor que R\$ 20,00; caso contrário, o lucro será de 30%. Escreva um programa que receba o valor do produto e exiba o valor da venda.

- 5) Escreva um programa que faz a conversão de temperatura de Celsius para Farenheit e de Farenheit para Celsius. Sugestão: utilize um caracter para definir em que unidade está a temperatura dada pelo usuário.
- 6) As maçãs custam R\$ 1,30 cada se forem compradas menos de uma dúzia, e R\$ 1,00 se forem compradas pelo menos 12. Escreva um programa que leia o número de maçãs compradas, calcule e escreva o custo total da compra.

## ATIVIDADE DE LABORATÓRIO

---

Exercícios propostos:

- 1) Faça um programa que leia um número e mostre uma mensagem indicando se este número é positivo ou negativo.
- 2) Faça um programa que apresenta o maior de dois números lidos do usuário.
- 3) Faça um programa que coloque dois nomes em ordem alfabética.
- 4) Faça um programa que apresente se o número que o usuário digitou é divisível por 3 e por 5 ao mesmo tempo.
- 5) Faça um programa que receba um ano (quatro dígitos) e informe se é um ano bissexto ou não. Pesquise quais as regras para o número ser bissexto.
- 6) Elabore um programa que leia do teclado o sexo de uma pessoa. Se o sexo digitado for “M” ou “m” ou “F” ou “f”, escrever na tela “Sexo válido!”. Caso contrário, exibir “Sexo inválido!”.
- 7) Num determinado Estado, para transferências de veículos, o DETRAN cobra uma taxa de 2,5% para carros fabricados antes de 2010 e uma taxa de 3,5% para os fabricados de 2010 em diante, taxa esta incidindo sobre o valor de tabela do carro. Escreva um programa lê o ano e o preço do carro e a seguir calcula e imprime a taxa a ser paga.

## EXERCÍCIOS EXTRAS

---

- 1) Quais das seguintes expressões são avaliadas como True?  
(a) `10 >= 8`    (b) `8 <= 10`    (c) `10 == 8`    (d) `10 != 8`    (e) `'8' < '10'`
- 2) Quais das seguintes expressões são avaliadas como True?  
(a) `'Dave' < 'Ed'`    (b) `'dave' < 'Ed'`    (c) `'Dave' < 'Dale'`



3) Qual é o valor da variável num após a seguinte execução?

```
>>> num = 10
>>> num = num + 5
>>> num == 20
>>> num = num + 1
```

4) Avalie as seguintes expressões lógicas usando as regras de precedência dos operadores:

(a)  $10 \geq 8$  and  $5 \neq 3$                       (b)  $10 \geq 8$  and  $5 == 3$  or  $14 < 5$

5) Qual das seguintes expressões lógicas não é logicamente equivalente às outras duas?

(a) `not (num < 0 or num > 10)`

(b) `num > 0 and num < 10`

(c) `num >= 0 and num <= 10`

6) Determine os resultados obtidos na avaliação das expressões lógicas seguintes sabendo que **a**, **b**, **c** contém, respectivamente, **2**, **7**, **3.5**, e que existem duas variáveis **m** e **n** cujos conteúdos são, respectivamente, **False** e **True**:

a) `b == a * c and (m or n)`

b) `b > a or b == math.pow(a, a)`

c) `m and b // a >= c or not a <= c`

d) `not m or n and math.sqrt(a + b) >= c`

e) `b/a == c or b/a != c`

f) `m or math.pow(b, a) <= c * 10 + a * b`

7) Escreva um programa que leia um número e apresente a raiz quadrada caso seja positivo ou nulo, e o quadrado do número caso seja negativo.

8) Elabore um programa que leia dois números reais e mostre o resultado da diferença do maior valor pelo menor.

9) Escreva um programa que leia um número inteiro de 3 dígitos e imprima se o algarismo da dezena é par ou ímpar.

10) Elabore um programa que leia notas de três avaliações de um aluno. A primeira avaliação tem peso 2, a segunda tem peso 3 e, a terceira, peso 5. Calcule a média do aluno. Se a média do aluno for maior ou igual a 6, o aluno está aprovado; caso contrário, o aluno está reprovado. Mostre o resultado da decisão.

11) Um pescador comprou um computador 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. Escreva um programa que leia o peso de peixes, e verifique se há excesso. Se houver, determine o peso excedente e o valor da multa. Caso contrário, mostrar "Dentro do regulamento".