

Área de Ciências Exatas e Engenharias
Lógica Computacional e Programação

Comandos Sequenciais: Teste de mesa

Professora:
Helena Graziottin Ribeiro

Teste de Mesa

- Um programa executa uma sequência de instruções que você desenvolveu
- Então **você** pode **testar** a execução do seu programa sem o computador!

Como assim?

Teste de Mesa

- Exemplo: qual o valor final das variáveis **a** e **b**?

```
a=5  
b=10  
a=a+b  
b=a+b  
a=a+b  
b=a+b  
a=a+b  
b=a+b
```

Teste de Mesa

- Criar uma tabela com as variáveis, e realizar cada instrução, linha a linha:

=> **a=5**

b=10

a=a+b

b=a+b

a=a+b

b=a+b

a=a+b

b=a+b

a	b
5	

Teste de Mesa

- Criar uma tabela com as variáveis, e realizar cada instrução, linha a linha:

=> $a=5$
 $b=10$
 $a=a+b$
 $b=a+b$
 $a=a+b$
 $b=a+b$

a	b
5	10

Teste de Mesa

- Criar uma tabela com as variáveis, e realizar cada instrução, linha a linha:

=> `a=5`
`b=10`
`a=a+b`
`b=a+b`
`a=a+b`
`b=a+b`

a	b
5	10
15	

Teste de Mesa

- Criar uma tabela com as variáveis, e realizar cada instrução, linha a linha:

a=5

b=10

a=a+b

=> b=a+b

a=a+b

b=a+b

a=a+b

b=a+b

a	b
5	10
15	25

Teste de Mesa

- Criar uma tabela com as variáveis, e realizar cada instrução, linha a linha:

a=5

b=10

a=a+b

b=a+b

=> a=a+b

b=a+b

a=a+b

b=a+b

a	b
5	10
15	25
40	

Teste de Mesa

- Criar uma tabela com as variáveis, e realizar cada instrução, linha a linha:

a=5

b=10

a=a+b

b=a+b

a=a+b

=> **b=a+b**

a=a+b

b=a+b

a	b
5	10
15	25
40	65

Teste de Mesa

- Criar uma tabela com as variáveis, e realizar cada instrução, linha a linha:

a=5

b=10

a=a+b

b=a+b

a=a+b

b=a+b

=> a=a+b
b=a+b

a	b
5	10
15	25
40	65
105	

Teste de Mesa

- Criar uma tabela com as variáveis, e realizar cada instrução, linha a linha:

a=5

b=10

a=a+b

b=a+b

a=a+b

b=a+b

a=a+b

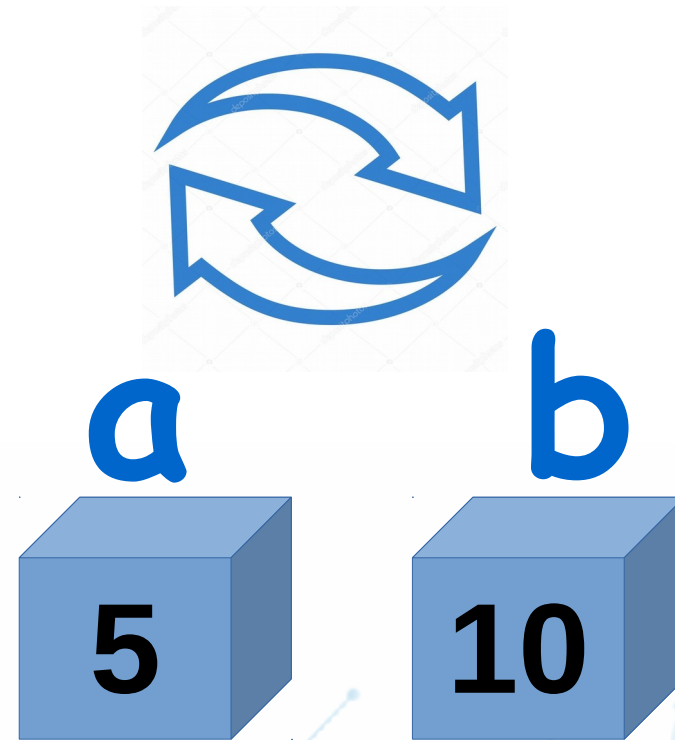
=> b=a+b

a	b
5	10
15	25
40	85
105	170

Problema

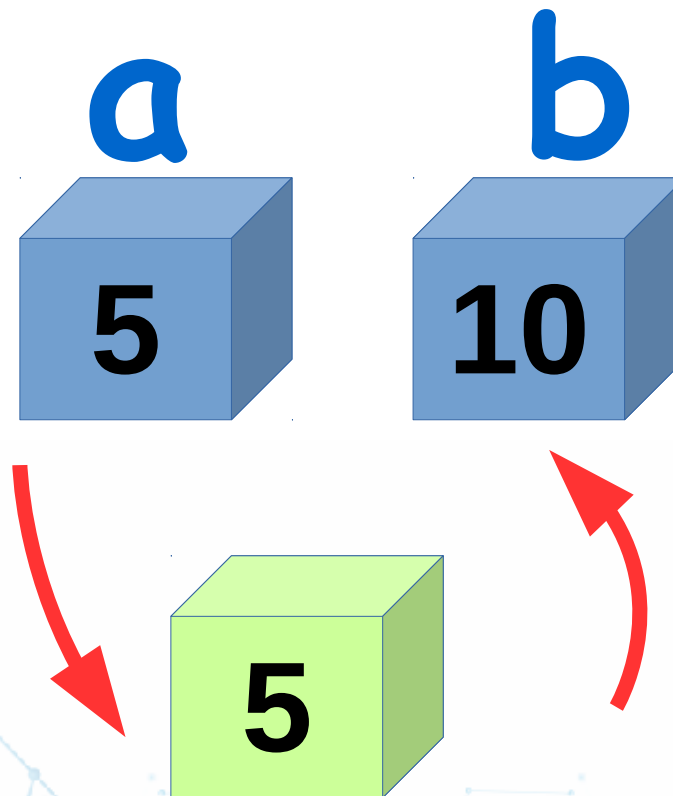
Como trocar o valor das variáveis?

Como fazer **a** ter o valor de **b**, e **b** ter o valor de **a**?



Problema

Utilizar uma variável auxiliar!



Problema

Utilizar uma variável auxiliar!

```
a=5  
b=10  
  
aux=a  
a=b  
b=aux  
  
print(a,b)
```

Problema

Utilizar uma variável auxiliar!

```
a=5  
b=10  
  
aux=a  
a=b  
b=aux  
  
print(a,b)
```

Fazer o
teste de mesa!

Problema

O meu programa deve ler um número qualquer, inteiro, e escrever uma mensagem dizendo se o número é par ou é ímpar.



Acho que preciso
fazer um teste...

Área de Ciências Exatas e Engenharias Lógica Computacional e Programação

Comando Condicional

Professora:
Helena Graziottin Ribeiro

Tipos de Dados

- Os computadores podem trabalhar com diversos tipos de dados.
- Os tipos de dados mais comuns são:
 - Dados numéricos inteiros (**int**)
 - Dados numéricos reais (**float**)
 - Dados literais (**strings**)
 - Dados lógicos:
- **Dados lógicos ou booleanos:**
 - Representa os dois únicos valores lógicos possíveis: VERDADEIRO e FALSO: **True** / **False**

Um **teste** é uma expressão que tem como resultado **VERDADEIRO** ou **FALSO**:

$x > 0$?

$a1 == 15$?

Comando if...else

Para fazer testes nos programas utiliza-se o comando **if**:

```
if <teste> :  
    <uma ou mais instruções>  
else :  
    <uma ou mais instruções>
```

Comando if...else

Para fazer testes nos programas utiliza-se o comando **if**:

```
if <teste> :  
    <uma ou mais instruções>  
else :  
    <uma ou mais instruções>
```

Instruções
executadas caso o
teste seja **verdadeiro**

Comando if...else

Para fazer testes nos programas utiliza-se o comando **if**:

```
if <teste> :  
    <uma ou mais instruções>  
else :  
    <uma ou mais instruções>
```

→ Instruções
executadas caso o
teste seja **falso**

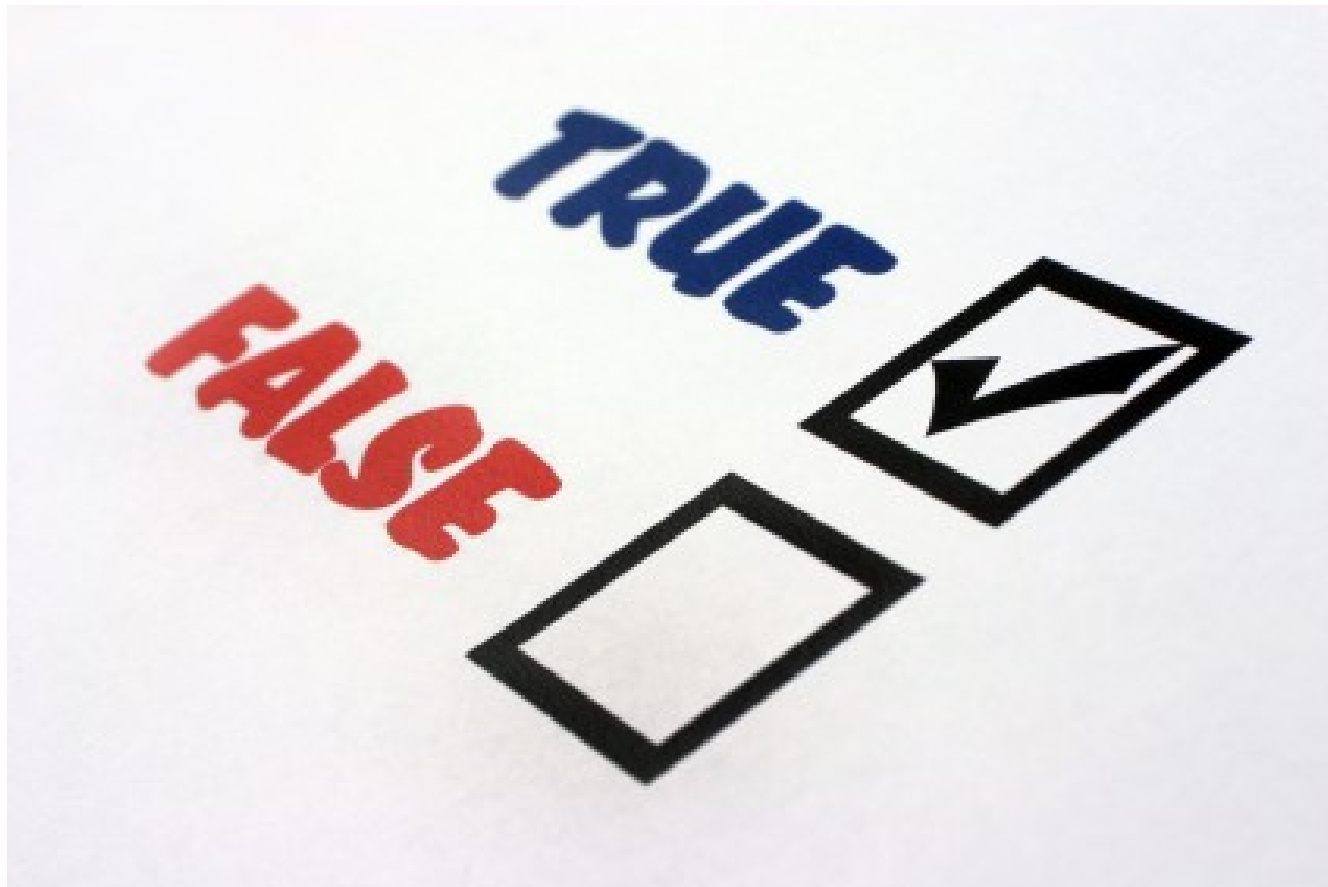
Comando if...else

Voltando ao problema:

O meu programa deve ler um número qualquer, inteiro, e escrever uma mensagem dizendo se o número é par ou se é ímpar:

```
x=(int)(input("numero="))  
if x%2 :  
    print("eh par!")  
else :  
    print("eh impar")
```


Operadores Relacionais



Operadores Relacionais

Operação	Operador
Igual	<code>==</code>
Diferente	<code>!=</code>
Maior	<code>></code>
Maior ou igual	<code>>=</code>
Menor	<code><</code>
Menor ou igual	<code><=</code>