

Exercícios sobre estruturas condicionais e precedência de operadores

Sumário

- [Exercício 7 - Maior valor](#)
- [Exercício 8 - Maior de três](#)
- [Precedência de operadores](#)
- [Exercício 9 - Polinômio](#)

Exercício 7 - Maior

- Faça um programa que receba como entrada dois valores, e imprima o maior valor entre eles. O programa deve executar as seguintes tarefas:
 - ENTRADA:
 1. Primeiro número
 2. Segundo número
 - SAÍDA:
 1. Maior número dentre os valores de entrada

Solução

```
val1 = int(input("Digite o primeiro valor: "))
val2 = int(input("Digite o segundo valor: "))

if val1>val2:
    print(f"Maior valor: {val1}")
else:
    print(f"Maior valor: {val2}")
```

Solução 2

Podemos também definir uma variável contendo o maior valor e imprimi-la no final

```
val1 = int(input("Digite o primeiro valor: "))
val2 = int(input("Digite o segundo valor: "))

if val1 > val2:
    maior_val = val1
else:
    maior_val = val2

print(f"Maior valor: {maior_val}")
```

Exercício 8 - Maior de três

- Faça um programa que receba como entrada 3 números e imprima na tela o maior número entre eles. Você pode considerar que os três números de entrada nunca serão iguais.
 - ENTRADA:
 1. Primeiro número
 2. Segundo número
 3. Terceiro número
 - SAÍDA:
 1. Maior número dentre os valores de entrada

Solução 1

```
val1 = int(input("Digite o primeiro valor: "))
val2 = int(input("Digite o segundo valor: "))
val3 = int(input("Digite o terceiro valor: "))

if val1 >= val2 and val1 >= val3:
    maior_val = val1
if val2 >= val1 and val2 >= val3:
    maior_val = val2
if val3 >= val1 and val3 >= val2:
    maior_val = val3

print(f"Maior valor: {maior_val}")
```

Solução 2

Não precisamos da terceira condição se usarmos a estrutura if...elif...else

```
val1 = int(input("Digite o primeiro valor: "))
val2 = int(input("Digite o segundo valor: "))
val3 = int(input("Digite o terceiro valor: "))

if val1 >= val2 and val1 >= val3:
    maior_val = val1
elif val2 >= val1 and val2 >= val3:
    maior_val = val2
else:
    maior_val = val3

print(f"Maior valor: {maior_val}")
```


Solução 3

Podemos também resolver o problema da seguinte forma:

```
val1 = int(input("Digite o primeiro valor: "))
val2 = int(input("Digite o segundo valor: "))
val3 = int(input("Digite o terceiro valor: "))

if val1 >= val2:
    maior_val = val1
else:
    maior_val = val2

if val3 > maior_val:
    maior_val = val3

print(f"Maior valor: {maior_val}")
```

Solução 4

Mas provavelmente a solução mais simples é

```
val1 = int(input("Digite o primeiro valor: "))
val2 = int(input("Digite o segundo valor: "))
val3 = int(input("Digite o terceiro valor: "))

maior_val = val1
if val2 > maior_val:
    maior_val = val2
if val3 > maior_val:
    maior_val = val3

print(f"Maior valor: {maior_val}")
```

Precedência de operadores

- Em uma expressão aritmética, os operadores + - * / e ** são executados em uma ordem específica;
- O operador ** (potência) sempre é executado primeiro. Portanto dizemos que ele possui a maior precedência;
- Os operadores * (multiplicação) e / (divisão) são executados após a potência e antes das somas e subtrações;
- Portanto, temos a seguinte tabela de precedência:

Operador	Precedência entre os operadores + - * / e **
**	1
* e /	2
+ e -	3

- Se dois operadores com a mesma precedência forem utilizados, eles são avaliados da esquerda para a direita, isto é, o operador mais à esquerda é avaliado primeiro.

Precedência de operadores

Sempre que tivermos dúvida sobre a precedência de uma operação, podemos utilizar parênteses () para garantir que a operação será realizada na ordem esperada.

Alguns exemplos:

Expressão	Resultado	Equivalente a
$2-3*4$	-10	$2-(3*4)$
$2/2*3$	3.0	$(2/2)*3$
$2**2*4$	16	$(2**2)*4$
$(2-3)*4$	-4	-----
$-2**2$	-4	$-(2**2)$
$(-2)**2$	4	$-2*-2$

Precedência de operadores

A precedência não é válida somente para operadores aritméticos. Todos os operadores da linguagem possuem uma precedência:

Operador	Descrição
(), []	
**	
+x, -x	Exemplo: $4 * -5 = -20$
*, /, %	% é o resto da divisão entre dois números
+, -	
&	AND bit a bit
^	XOR bit a bit
	OR bit a bit
<, <=, >, >=, !=, ==	
not x	not inverte uma expressão entre True e False
and	
or	

Exercício 9 - Polinômio

- Faça um programa que encontre as raízes de um polinômio do segundo grau utilizando a fórmula de Bhaskara

$$ax^2 + bx + c = 0 \qquad \tilde{x} = \frac{-b \pm \sqrt{b^2 - 4a * c}}{2a}$$

- ENTRADA:
 1. Coeficientes **a**, **b** e **c**
- SAÍDA:
 1. Raízes do polinômio (valores de **x** que resolvem a equação)
- Lembrem-se:
 - Se $b^2 - 4a * c < 0$, não há raízes reais
 - Se $b^2 - 4a * c = 0$, há apenas uma raiz válida
 - Se $a = 0$, há apenas uma raiz dada por $\tilde{x} = -c/b$
- Para calcular a raiz quadrada:
 - $\text{raiz} = \text{valor}^{**0.5}$ (elevar um número a 0.5 é equivalente a calcular a raiz)

Solução

```
# Coeficientes do polinômio  $ax^2+bx+c$ 
a = float(input("Coeficiente a: "))
b = float(input("Coeficiente b: "))
c = float(input("Coeficiente c: "))

if a==0:
    if b==0:
        print("Não é um polinômio pois a e b são 0!")
    else:
        # Polinômio do tipo  $bx+c=0$  pois  $a==0$ 
        x1 = -c/b
        print(f"Existe uma raiz: {x1}")
else:
    delta = b**2-4*a*c
    if delta<0:
        print("Não existem raízes reais")
    elif delta==0:
        x1 = -b / (2*a)
        print(f"Existe uma raiz: {x1}")
    else:
        x1 = (-b + delta**0.5) / (2*a)
        x2 = (-b - delta**0.5) / (2*a)
        print(f"Raiz 1: {x1}")
        print(f"Raiz 2: {x2}")
```