

SISB093 - Programação 3

Aula 4

Prof. Thiago Cavalcante

Divisão pelo piso e módulo

- Divisão pelo piso = Quociente da divisão `//`
- Módulo = Resto da divisão `%`

Expressões booleanas

- Expressões cujo resultado é `True` ou `False`
- Produzidas com um grupo diferente de operadores (relacionais)

Operadores lógicos

- `and` , `or` e `not` (`e` , `ou` e `não`)
- São utilizados em conjunto com expressões booleanas

Execução condicional

```
if condição:  
    # sequência de instruções caso  
    # a condição seja verdadeira
```

Execução alternativa

```
if condição:  
    # sequência de instruções caso  
    # a condição seja verdadeira  
else:  
    # sequência de instruções caso  
    # a condição seja falsa
```

Condicionais encadeadas

```
if primeira condição:
    # sequência de instruções caso
    # a primeira condição seja verdadeira
elif segunda condição:
    # sequência de instruções caso
    # a primeira condição seja falsa e
    # a segunda condição seja verdadeira
# ...
# sequência de elifs
# ...
else: # opcional
    # sequência de instruções caso
    # todas as condições anteriores
    # sejam falsas
```

Condicionais aninhadas

```
if primeira condição:  
    # sequência de instruções caso  
    # a primeira condição seja verdadeira  
else:  
    if segunda condição:  
        # sequência de instruções caso  
        # a primeira condição seja falsa  
        # e a segunda condição seja verdadeira  
    else:  
        # sequência de instruções caso  
        # a primeira condição seja falsa  
        # e a segunda condição seja falsa
```


Entrada de teclado

Função `input`

Exercício 5.1

O módulo `time` fornece uma função, também chamada `time`, que devolve a Hora Média de Greenwich na "época", que é um momento arbitrário usado como ponto de referência. Em sistemas UNIX, a época é primeiro de janeiro de 1970.

```
>>> import time
>>> time.time()
1437746094.5735958
```

Escreva um script que leia a hora atual e a converta em um tempo em horas, minutos e segundos, mais o número de dias desde a época.

Exercício 5.3

1. Escreva uma função chamada `checar_triangulo` que receba três números inteiros como argumentos, e que imprima "Sim" ou "Não", dependendo da possibilidade de formar ou não um triângulo de gravetos com os comprimentos dados.
2. Escreva uma função que peça ao usuário para digitar três comprimentos de gravetos, os converta em números inteiros e use `checar_triangulo` para verificar se os gravetos com os comprimentos dados podem formar um triângulo.