

Universidade Federal de Viçosa
Departamento de Informática
Centro de Ciências Exatas e Tecnológicas



INF 100 – Introdução à Programação

Comandos de Repetição
para ou “para cada” (*for*)

Aula Prática 03

soma = 0 *# essa variável armazenará a soma dos pesos válidos lidos*
n = 0 *# essa variável armazenará a quantidade de pesos válidos lidos*

```
peso = float (input('Peso da capivara 1: '))  
if peso > 0:  
    soma = soma + peso  
    n = n + 1  
else:  
    print('Peso inválido!')
```

```
peso = float (input('Peso da capivara 2: '))  
if peso > 0:  
    soma = soma + peso  
    n = n + 1  
else:  
    print('Peso inválido!')
```

```
peso = float (input('Peso da capivara 3: '))  
if peso > 0:  
    soma = soma + peso  
    n = n + 1  
else:  
    print('Peso inválido!')
```

```
peso = float (input('Peso da capivara 4: '))  
if peso > 0:  
    soma = soma + peso  
    n = n + 1  
else:  
    print('Peso inválido!')
```

```
if n == 0:  
    print('\nNenhum peso válido fornecido.')  
else:  
    print('\nPeso médio = %5.1f kg' % (soma/n) )  
    print('(considerando %d pesos válidos)' % n )
```

Aula Prática 03

```
soma = 0
n = 0

i = 1
while i <= 4:
    peso = float (input('Peso da capivara %d: ' % (i)))
    if peso > 0:
        soma = soma + peso
        n = n + 1
    else:
        print('Peso inválido!')
    i = i + 1

if n == 0:
    print('\nNenhum peso válido fornecido.')
else:
    print('\nPeso médio = %5.1f kg' % (soma/n) )
    print('(considerando %d pesos válidos)' % n )
```



Aula Prática 03

(Obrigar o usuário a digitar 4 pesos válidos)

```
soma = 0
n = 0

while n < 4:
    peso = float(input('Peso da capivara %d: ' % (n+1)))
    if peso > 0:
        soma = soma + peso
        n = n + 1
    else:
        print('Peso inválido!')

print('\nPeso médio = %5.1f kg' % (soma/4))
```



Aula Prática 03

(Obrigar o usuário a digitar 4 pesos válidos)

```
soma = 0
n = 0

while n < 4:
    peso = float(input('Peso da capivara %d: ' % (n+1)))
    if peso > 0:
        soma = soma + peso
        n = n + 1
    else:
        print('Peso inválido!')

print('\nPeso médio = %5.1f kg' % (soma/4))
```

O que esse *loop* faz é simplesmente contar de 0 a 3 (ou de 1 a 4)



Exercício "Números Primos"

- Faça um programa que pede para o usuário digitar um número e, então, testa se o número digitado é um número primo.
- Adapte este programa para que ele possa testar a primalidade de vários números.
- Use o número 0 como 'sentinela' para indicar o término do programa.



Números Primos

(solução mais eficiente)

```
n = int( input("Entre com o valor de N (> 0): "))
if n == 2:  # único número par primo
    EhPrimo = True
elif n == 1 or n % 2 == 0:  # se n for 1 ou par, não precisa verificar
    EhPrimo = False
else:
    EhPrimo = True
    # Tenta dividir n por x = 3 até raiz( n )+1, pegando só os ímpares
    x = 3
    limite = int( n ** 0.5 ) + 1
    while EhPrimo and x <= limite:
        if n % x == 0:
            EhPrimo = False
        x = x + 2
if EhPrimo: print( n, "é primo.\n")
else: print( n, "não é primo.\n")
```



Números Primos

(solução mais eficiente)

```
x = 3
limite = int( n ** 0.5 ) + 1
while EhPrimo and x <= limite:
    if n % x == 0:
        EhPrimo = False
    x = x + 2
```

Esse *loop* também é uma contagem simples, fazendo x iniciar em 3 e ir incrementando de 2 em 2:

3, 5, 7, ...

até atingir um limite ou até encontrar um divisor para n.



Comando “para” ou “para cada”

```
para i = valor_inicial até valor_final:  
    (faça alguma coisa)  
fim_para
```

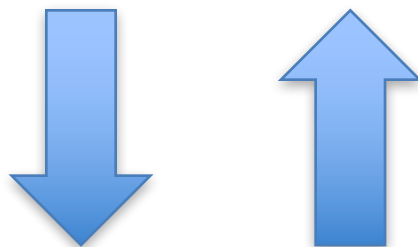


```
i = valor_inicial  
enquanto i ≤ Valor_final:  
    (faça alguma coisa)  
    i = i + 1  
fim_enquanto
```



Comando “para” ou “para cada”

```
para i = valor_inicial até valor_final:  
    Escreva i  
fim_para
```



```
i = valor_inicial  
enquanto i ≤ Valor_final:  
    Escreva i  
    i = i + 1  
fim_enquanto
```



Comando “para” ou “para cada”

```
para i = 1 a 5:  
    Escreva i  
fim_para
```

Exemplo



Resultado

```
1  
2  
3  
4  
5
```



Aula Prática 03

(Obrigar o usuário a digitar 4 pesos válidos)

```
soma = 0
n = 0

while n < 4:
    peso = float (input('Peso da capivara %d: ' % ( n+1 )))
    if peso > 0:
        soma = soma + peso
        n = n + 1
    else:
        print('Peso inválido!')

print('\nPeso médio = %5.1f kg' % (soma/4) )
```



Aula Prática 03

Usando o comando “para”...

soma = 0

para i = 1 a 4:

 (Leia um peso válido)

 soma = soma + peso

fim_para

Escreva soma/4



Aula Prática 03

Usando o comando “para”...

```
soma = 0
```

```
para i = 1 a 4:
```

```
    Leia peso
```

```
    enquanto peso < 0:
```

```
        Escreva “Peso inválido!”
```

```
        Leia peso
```

```
    fim_enquanto
```

```
    soma = soma + peso
```

```
fim_para
```

```
Escreva soma/4
```

Leia um peso válido



Comando “for” em Python

```
para i = 1 a 5:  
  Escreva i  
fim_para
```

(algoritmo)

```
for i in range(1, 6):  
  print( i )
```

(Python)

Em Python,
precisamos somar 1
ao “valor final”



Aula Prática 03

(Obrigar o usuário a digitar 4 pesos válidos)

```
soma = 0
```

```
for i in range( 1, 5 ):
    # Leia peso > 0
    peso = float (input('Peso da capivara %d: ' % (i)))
    while peso <= 0:
        print('Peso inválido!')
        peso = float (input('Peso da capivara %d: ' % (i)))
    soma = soma + peso

print('\nPeso médio = %5.1f kg' % (soma/4) )
```



Aula Prática 03

(Estendendo para um número qualquer de capivaras)

```
n = int (input('Entre com o número de capivaras: '))

soma = 0

for i in range( 1, n+1 ):
    # Leia peso > 0
    peso = float (input('Peso da capivara %d: ' % (i)))
    while peso <= 0:
        print('Peso inválido!')
        peso = float (input('Peso da capivara %d: ' % (i)))
    soma = soma + peso

print('\nPeso médio = %5.1f kg' % (soma/n) )
```



Números Primos

(solução mais eficiente)

```
x = 3
limite = int( n ** 0.5 ) + 1
while EhPrimo and x <= limite:
    if n % x == 0:
        EhPrimo = False
    x = x + 2
```

Esse *loop* também é uma contagem simples, fazendo x iniciar em 3 e ir incrementando de 2 em 2:

3, 5, 7, ...

até atingir um limite ou até encontrar um divisor para n.



Comando “para” com passo $\neq 1$

```
para i = 1 a 10, passo 2:  
    Escreva i  
fim_para
```

Exemplo



Resultado

1
3
5
7
9



Comando “para” com passo $\neq 1$

```
para i = 1 a 5, passo 2:  
  Escreva i  
fim_para
```

(algoritmo)

```
for i in range(1, 6, 2):  
  print( i )
```

(Python)

1
3
5

Resultado



Números Primos

(solução mais eficiente)

```
x = 3
limite = int( n ** 0.5 ) + 1
while EhPrimo and x <= limite:
    if n % x == 0:
        EhPrimo = False
    x = x + 2
```

```
limite = int( n ** 0.5 ) + 1
for x in range( 3, limite+1, 2 ):
    if n % x == 0:
        EhPrimo = False
    break
```



Números Primos

(usando o comando 'for')

```
n = int( input("Entre com o valor de N (> 0): "))
if n == 2:  # único número par primo
    EhPrimo = True
elif n == 1 or n % 2 == 0:  # se n for 1 ou par, não precisa verificar
    EhPrimo = False
else:
    EhPrimo = True
    # Tenta dividir n por x = 3 até raiz( n )+1, pegando só os ímpares
    limite = int( n ** 0.5 ) + 1
    for x in range( 3, limite+1, 2 ):
        if n % x == 0:
            EhPrimo = False
            break

if EhPrimo: print( n, "é primo.\n")
else: print( n, "não é primo.\n")
```



Exercício "Triângulo X"

Escreva um programa em Python que peça ao usuário o valor de um inteiro $n > 0$ e então desenhe um triângulo retângulo, usando o caractere "X", com altura e base iguais a n . Não é necessário que o programa verifique se o valor de n é válido ou não. Veja abaixo um exemplo da execução do programa: a saída para a entrada de $n = 5$ é um triângulo retângulo de altura e base iguais a 5.



Exercício "Triângulo X"

Entre com o valor de n: 5

X

XX

XXX

XXXX

XXXXX

