

# 4 Aula 04: 10/SET/2020



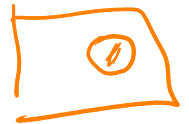
## 4.1 Aulas passadas

- variáveis
- comando de atribuição
- função de escrita na tela (= `print()`)
- função de leitura via teclado (= `input()`)
- tipos `str`, `int`, `bool` e `float`
- funções de conversão `str()`, `int()`, `bool()` e `float()`
- f-strings
- expressões e operadores relacionais
- Spyder
- comando de repetição `while`
- contador de iterações: imprime primeiros impares

De composição  
Parabéns while

3.14 2.7

True



`0 == 0`  
`1 =`  
`> =`  
`< =`  
`> <`

`+`  
`-`  
`*`  
`/`

`**`

## 4.2 Avisos

- EP01 deve ser entregue até amanhã, 14/09
- EP02 deve ser entregue até amanhã, 21/09
- p02 estará disponível de amanhã às 12h até sábado às 23h59m
- Não deixe de visitar Haile, toda segunda às 13h.

←  
→ vania, tipos  
expresso

→ Meet

### 4.3 Avisos sobre o EP01

- avaliação é divulgada **imediatamente após** a submissão
  - há quem entregou o esqueleto do EP...
  - programa deve ter **apenas 2** `input()`
  - **siga a especificação do enunciado**
  - teste com os exemplos do enunciado
  - `'soma = -113'` é diferente de `'Soma = -113'`
  - **não remova ou altere** o cabeçalho
  - escreva seu programa **depois do comentário**
  - **#** escreva a seguir a sua solução
  - aqueles que não conseguiram instalar **Anaconda** ou algum IDE de Python como `idle` pode tentar usar `https://repl.it`
- Handwritten notes:*
- Red arrows point to "imediatamente após a submissão" and "há quem entregou o esqueleto do EP...".
  - Red text "Arquivo" with an arrow pointing to the top right.
  - Green circles containing "a x" and "b y" with arrows pointing to "apenas 2 input()".
  - Green circles around "soma = -113" and "Soma = -113" with an arrow pointing to "'s' ≠ 'S'".
  - Green circles around "siga a especificação do enunciado" and "teste com os exemplos do enunciado".
  - Pink circles around "não remova ou altere o cabeçalho" and "# escreva a seguir a sua solução".
  - Pink circle around "https://repl.it" with an arrow pointing to the right.

## 4.5 Exercício: potências

Dados números inteiros  $n$  e  $k$  com  $k \geq 0$ , determinar  $n^k$ . Por exemplo, dados os números 3 e 4 o seu programa deve escrever o número 81.

### 4.5.1 Exemplos

Digite o valor de  $n$ : 3

Digite o valor de  $k$ : 4

3 elevado a 4 é 81

Digite o valor de  $n$ : 12

Digite o valor de  $k$ : 5

12 elevado a 5 é 248832

Digite o valor de  $n$ : -3

Digite o valor de  $k$ : 21

-3 elevado a 21 é -10460353203

← escrever print()

while enquanto

$n$   
↓  
2

$k$   
↓  
3

pot  
↓  
?

mas?

$2 * 2 * 2 * 2$

$pot = 2 * 2 * \dots * 2$

$((2 * 2) * 2) * 2) * 2) * 2) * 2$

potencia parcial \* 2

$n$  ↓ 2  
 $k$  ↓ 3  
 $pot$  ↓ 8  
↓ 1  
↓ 2  
↓ 4

$pot = pot * 2$   
 $pot \leftarrow pot * n$

$pot \leftarrow 1$   
 $pot \leftarrow pot * n$   
 $pot \leftarrow pot * n$

~~pot = 4~~

~~$pot \leftarrow pot * n * 2$~~

4.5.2 Solução 11211 Dados: 2 3

ingratos

```

n = int(input("Digite o valor de n: "))
k = int(input("Digite o valor de k: "))
# print("O valor da potência é =", n**k)
pot = 1
i = 0
while i < k:
    pot = pot*n
    i = i + 1
print(f"{n} elevado a {k} é {pot}")

```

nativo

n	k	pot	i
2	3	7	0
		2	1
		4	2
		8	3

↓  
 2 elevado a 3 é 8

## 4.11 Exercício: fatorial

$$n! = 1 \times 2 \times 3 \times \dots \times n$$

Dado um inteiro não negativo  $n$ , determinar  $n!$

ler

escreve

### 4.11.1 Exemplos

Digite o valor de  $n$ : 3

$$3! = 6$$

$$3 \times 2 \times 1 = 1 \times 2 \times 3 = 6$$

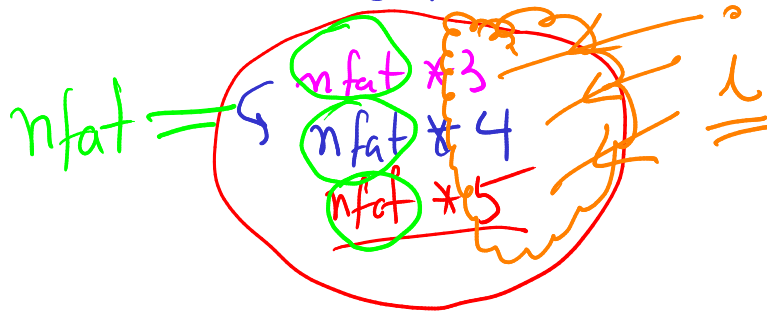
Digite o valor de  $n$ : 15

$$15! = 1307674368000$$

$$1 \times 2 \times 3 \dots \times 15$$
$$\left( (1 \times 2) \times 3 \right) \times 4 \times \dots \times 15 \times 100$$

Digite o valor de  $n$ : 6

$$6! = 720$$



## 4.11.2 Solução

```
'''
Programa que lê um inteiro não negativo n, e calcula n!
'''

print("Cálculo do fatorial de um número\n")

# leia o valor de n
n = int(input("Digite um número inteiro não-negativo: "))

# inicializações
i = 1 # contador
n_fat = 1

# calcule n!
while i <= n:
    n_fat = n_fat * i
    i = i + 1

print(f"{n}! = {n_fat}")
```