



DEPARTAMENTO
DE INFORMÁTICA
PUC-RIO

 Open in Colab

Roteiro da primeira lista de exercícios

Ao longo deste documento, haverá vários exemplos e exercícios comentados para consolidar o conhecimento adquirido sobre tipos de dados, operadores aritméticos e expressões aritméticas. Ao final, serão apresentados alguns desafios para os quais as respostas são fornecidas para simples conferência.

Operadores aritméticos e expressões

```
In [ ]: # Este é um comentário. Não é considerado pelo ambiente na execução  
# Execute esta célula que contém a expressão abaixo com dois valores  
# Repare que é utilizado o ponto decimal.  
18.72 + 3.34 # É possível escrever um comentário ao lado de uma
```

```
In [ ]: # Mas cuidado! Ao executar o código abaixo, ocorrerá um erro.  
# Motivo: um espaço em branco antes da expressão.  
18.72 + 3.34
```

Agora, execute as próximas 7 células com expressões envolvendo apenas valores inteiros. Certifique-se de analisar com cuidado os 3 últimos resultados.

```
In [ ]: 13 + 5
```

```
In [ ]: 13 - 5
```

```
In [ ]: 13 * 5
```

```
In [ ]: 13 ** 5
```

```
In [ ]: 13 / 5
```

O operador `/` é responsável por executar a operação de divisão como conhecemos na Matemática. Por isso, o resultado de 13 dividido por 5 tem como resultado 2.6.

In []: `13 // 5`

Já o operador `//` executa uma divisão na qual o resultado é o maior inteiro menor do que o quociente da divisão tradicional. Na expressão acima, o resultado é 2 porque este é o maior inteiro menor que 2.6 (resultado de $13 / 5$).

In []: `13 % 5`

O operador `%` calcula o resto da divisão do primeiro operando pelo segundo. No caso, o resultado é 3: $13 - (13 // 5) * 5$.

In []: `# Execute para conferir se gera o mesmo resultado que 13 % 5`
`13 - (13 // 5) * 5`

Mas cuidado! A expressão abaixo gera um valor que pode parecer estranho para alguns. Lembre-se que o resultado é o maior inteiro menor do que o quociente da divisão tradicional (-2.6).

In []: `(-13) // 5`

Calcule o resultado da expressão aritmética abaixo.

In []: `10 - 3 * 2 + 4 ** 3 // 8`

O resultado já era conhecido pois é a mesma expressão vista no texto do início da primeira aula. Mas como fica a expressão com alguns parênteses? O resultado muda? Experimente!

In []: `(10 - 3) * 2 + 4 ** (3 // 8)`

Claro! Os parênteses alteram a precedência das operações, ou seja, mudam a ordem na qual as operações são executadas. Em primeiro lugar, são executadas as operações entre parênteses. Em seguida, a expressão acima passa a ser: $7 * 2 + 4 ** 0$, gerando como resultado o valor 15.

In []: `7 * 2 + 4 ** 0`

A execução do código da próxima célula provoca um erro comum, principalmente, para iniciantes na programação. Execute-a e observe a mensagem de erro exibida.

In []: `10 / 0` `# não é possível dividir um valor por zero!`

Seção de treino

Nesta seção, adicione novas células de código e tente criar suas variáveis e

expressões. Bom treino!

Desafios

Agora, resolva os desafios abaixo, adicionando novas células de código e definindo as expressões necessárias para calcular o resultado. Repare que a resposta de cada desafio é dada.

DESAFIO 1

Qual o volume, em metros cúbicos, de um reservatório, em forma de paralelepípedo, com 79 metros de comprimento, 10 metros de largura e 4 metros de profundidade?

Resposta: ao final deste documento.

DESAFIO 2

Considerando que 1 metro cúbico corresponde a 1000 litros de água, quantos litros cabem no reservatório do desafio 1?

Resposta: ao final deste documento.

DESAFIO 3

Considere uma torneira que despeja 75 litros de água por hora. Em quantos minutos a torneira encheria o reservatório dos desafios 1 e 2?

Resposta: ao final deste documento.