

## Aula Prática 6

---

Adriano César Machado Pereira  
João Guilherme Maia de Menezes

### Instruções para Submissão

Na aula prática de hoje, você terá que elaborar programas para resolver problemas diversos, conforme descrito abaixo. Cada uma das soluções deverá ser implementada em seu próprio arquivo com extensão `.py`. Por exemplo, a solução para o problema 1 deverá ser implementada em um arquivo chamado `problema1.py`, a solução para o problema 2 deverá ser implementada no arquivo `problema2.py` e assim por diante. Finalmente, submeta cada um dos arquivos pelo Moodle.

**Dica:** se você tiver problemas com caracteres especiais (caracteres com acentos, por exemplo), adicione a linha abaixo na primeira linha de todos os arquivos `*.py`

```
# -*- coding: utf-8 -*-
```

### Problema 1

Faça um programa que leia um endereço IP e imprima na tela se o mesmo é válido ou não. Um endereço IP possui o formato `x.y.z.w`, onde `x`, `y`, `z` e `w` são números no intervalo `[0, 255]`.

**Observação:** as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

**Exemplo de execução do programa:**

```
200.135.80.9  
Válido
```

**Exemplo de execução do programa:**

257.32.4.5  
Inválido

## Problema 2

Em uma competição de salto em distância, cada atleta tem direito a cinco saltos. No final da série de saltos de cada atleta, o melhor e o pior resultados são eliminados. O seu resultado fica sendo a média dos três valores restantes. Você deve fazer um programa que receba as cinco distâncias alcançadas pelo atleta em seus saltos. Em seguida, o programa deve exibir a média dos saltos conforme a descrição acima informada (retirar o melhor e o pior salto e depois calcular a média).

**Observação:** as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

**Exemplo de execução do programa:**

6.50  
6.10  
6.20  
5.40  
5.3  
Média: 5.90

## Problema 3

Faça um programa que leia 5 números inteiros e armazene-os numa lista. Armazene os números pares em uma lista e os números ímpares em uma outra lista. Imprima as três listas na tela.

**Observação:** as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

**Exemplo de execução do programa:**

1  
2  
3  
4  
5  
[1, 2, 3, 4, 5]  
[2, 4]  
[1, 3, 5]

**Exemplo de execução do programa:**

```
2
4
6
8
10
[2, 4, 6, 8, 10]
[2, 4, 6, 8, 10]
[]
```

## Problema 4

Faça um programa que receba a temperatura média de cada mês do ano e armazene-as em uma lista. Após isto, calcule e mostre a média anual das temperaturas e mostre todas as temperaturas acima da média anual.

**Observação:** as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

**Exemplo de execução do programa:**

```
1
2
3
4
5
6
7
8
9
10
11
12
Média: 6.50
7
8
9
10
11
12
```

## Problema 5

Escreva um programa que leia uma data no formato **DD/MM/AAAA** e imprima na tela a data no formato **MM/DD/AAAA**.

**Observação:** as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

**Exemplo de execução do programa:**

01/11/2018  
11/01/2018

## Problema 6

Escreva uma função chamada `tamanho_maior_string` que recebe como parâmetro uma lista com strings e retorna o tamanho da maior string na lista. Sua função não deve modificar a lista original.

**Observação 1:** O nome da função deve ser exatamente como especificado acima e seu arquivo deve obrigatoriamente se chamar `problema6.py`.

**Observação 2:** Seu programa deve conter apenas a função descrita acima. Você não precisa realizar a entrada e saída de dados (não precisa usar as funções `input()` e `print()`).

**Exemplo 1 de execução do programa:**

['um', 'isso', 'é']  
4

**Exemplo 2 de execução do programa:**

['Brasil', 'Argentina', 'Estados Unidos', 'Austrália', 'Emirados Árabes Unidos']  
22

## Problema 7

Escreva uma função chamada `nested_sum` que receba uma lista de listas de números inteiros e retorna a soma dos elementos de todas as listas aninhadas.

**Observação 1:** O nome da função deve ser exatamente como especificado acima e seu arquivo deve obrigatoriamente se chamar `problema7.py`.

**Observação 2:** Seu programa deve conter apenas a função descrita acima. Você não precisa realizar a entrada e saída de dados (não precisa usar as funções `input()` e `print()`).

**Exemplo 1 de execução do programa:**

[[1, 2], [3, 4]]  
Soma: 10

**Exemplo 2 de execução do programa:**

**[[1, 2], [3], [4, 5, 6]]**  
**Soma: 21**