

Computação I - Python

Laboratório 4

Seguindo com nossas boas práticas, para cada um dos exercícios a seguir:

- NOVA NOTAÇÃO: a partir de agora, podem aparecer na especificação do retorno desejado nos enunciados o nome de uma variável ou parâmetro entre os sinais `<` e `>`, como por exemplo `<var>`. Isso significa que espera-se o valor da variável neste lugar, e não seu identificador.
- antes de começar a escrever código, faça o estudo do problema e o planejamento de sua solução.
- lembre de botar a **documentação** direitinho, dizendo o que a função faz, quais suas entradas e qual o **tipo de dado** de cada entrada, bem como do valor de retorno da função; por exemplo, se sua função recebe dois números inteiros, nos parâmetros chamados *a* e *b* e retorna a divisão deles (possivelmente um número fracionário):

```
'''Calcula e retorna a divisão de a por b;  
int, int -> float'''
```

- escolha **nomes elucidativos** para suas funções e parâmetros;
- pense em **valores de teste** relevantes para testar sua função. Ela tem alguma resposta esperada para valores negativos? Valores fracionários? Que tal testar também com valores no extremo do conjunto de dados de interesse da função (maiores valores esperados, menores valores esperados)?
- quando estiver com dificuldade para entender algum erro de funcionamento ou resultado inadequado de sua função, não fique paralizado olhando para a tela! Pegue lápis e papel e recorra ao **teste de mesa**.
- para fazer a entrega desta atividade prática, escreva suas funções no editor do **IDLE**, salvando todas em um único arquivo.

Vamos lá!

1. Faça uma função em Python chamada **SIGA**, que receba uma tupla contendo 4 informações: o nome do aluno e três notas. Sua função deve retornar uma tupla, cujo primeiro elemento é o nome do aluno, o segundo elemento é a média, e o terceiro elemento é a situação do aluno, que deve ser representada por uma string. Se a média das três notas do aluno for maior ou igual a 7 (inclusive), a função deverá retornar: (`<nome>`, `<media>`, 'aprovado', 'Parabéns!'). Se a média do aluno for menor que 7, porém maior ou igual a 5, sua função deve retornar: (`<nome>`, `<media>`, 'aprovado'). Se a média for menor que 5, a função deve retornar (`<nome>`, `<media>`, 'reprovado'). Esta função deverá retornar a média com uma casa decimal apenas.

0	Macaco
1	Galo
2	Cão
3	Porco
4	Rato
5	Boi
6	Tigre
7	Coelho
8	Dragão
9	Serpente
10	Cavalo
11	Carneiro

2. O zodíaco chinês é composto por animais com ciclo de 12 anos. Uma maneira simplificada de identificá-lo, com uma pequena margem de erro, é verificando-se apenas o ano de nascimento de uma pessoa. Para isso, pegamos o ano de nascimento e dividimos por 12 e o signo aproximado será encontrado conforme o resto dessa divisão, de acordo com a tabela de signos.

Faça uma função que receba o ano de nascimento e retorne o signo aproximado correspondente. **Você consegue resolver este problema sem usar o IF?**

3. Atualmente no Brasil (ano 2021), números de telefone fixo tem 8 dígitos, e os números de telefone celular, 9 dígitos. Esse número é regional, então para completar a identificação do usuário no Brasil a que um número corresponde, é importante também o código de área (DDD), composto de 2 dígitos. É comum as pessoas fornecerem seu número telefônico tanto incluindo o código de área, quanto não incluindo. Seguindo apenas essas regras de tamanho, alguns exemplos de números de telefone válidos no Brasil:

21912316165
1132316165
71912316165
323231616
32316165
912316165

Faça uma função que receba como entrada uma string contendo uma sequência de dígitos, que supostamente corresponde ao número de telefone informado por um usuário (parta do princípio que essa string só tem dígitos). Sua função deve identificar se o número de telefone é um número válido no Brasil (ou seja, se se encaixa em um dos padrões aceitáveis). Caso seja, sua função deve retornar uma tupla com duas strings, a primeira contendo exatamente dois dígitos que correspondam ao DDD (ou string vazia, caso o DDD não tenha sido informado), e a segunda, contendo 8 ou 9 dígitos, correspondente ao número de telefone sem o DDD. Caso o número seja inválido, sua função deve retornar uma tupla com duas strings vazias.

4. Faça uma função em Python chamada **formato_data**, que receba uma string de 8 posições, referente a uma data, e retorna os possíveis formatos em que a data fornecida possa ser interpretada. Você pode assumir que a string de entrada contém duas barras ('/') nas posições 2 e 5, e que o resto dos caracteres estão no intervalo entre '0' e '9'. Cada formato é especificado como uma string e existem três tipos de formátos válidos: 'dd/mm/yy', 'yy/mm/dd' e 'mm/dd/yy'. Como uma data pode, em alguns casos, ser interpretada como pertencente a mais que um formato válido, você deve retornar uma tupla contendo todos os formatos válidos para a data de entrada. Caso nenhum formato case com os dados de entrada, você deve retornar uma tupla vazia. Para simplificar, você pode considerar que todos os meses possuem 31 dias e que tanto o ano, quanto o mês e o dia são sempre especificados por dois dígitos. Veja abaixo alguns exemplos de uso desta função na shell.

```
>>> formato_data('98/25/07')
()
>>> formato_data('01/01/00')
('dd/mm/yy', 'mm/dd/yy')
>>> formato_data('00/10/01')
('yy/mm/dd',)
>>> formato_data('01/01/01')
('dd/mm/yy', 'mm/dd/yy', 'yy/mm/dd')
```