

Computação I

Lista de exercícios 6 – Tuplas e strings.

Atenção! Leia as instruções antes de fazer a lista! A padronização do nome do arquivo e dos nomes das funções é muito importante e está explicada no arquivo de instruções!

Data de entrega: 26/10/2022

A menos que esteja explicitamente pedido na questão, não utilize nenhum método ou função já existente do Python exceto pelas funções print, len, format, e range. De forma simplificada, função é tudo o que precisa ser chamado com parênteses, e método é tudo o que precisa ser chamado com parênteses, mas que está associado a uma variável, isto é, variavel.metodo(args). Funções de transformação de tipo (int, float, str, list, tuple...) **não** são permitidas. Não importe nenhum módulo. Não é necessário testar se os dados passados por argumento são válidos. Não crie funções dentro de funções. **Atenção para a diferença entre o return e o print.** O return permite que o valor seja utilizado fora da função, enquanto o print só imprime na tela. Quando eu peço para retornar um valor (independentemente do tipo), vocês precisam utilizar o return; quando eu peço para mostrar, ou imprimir, na tela, vocês precisam usar o print. Outro detalhe é que a utilização de return print('string') está errada, pois nesse caso o valor retornado não é a string e sim None (o return está retornando o que o print retorna, que é None). Isso faz diferença na correção e, principalmente, é um conceito importante.

1. Escreva uma função em Python que recebe, por argumento, uma tupla com nomes de livros, uma tupla com os autores, uma tupla com a quantidade de páginas e uma tupla com o ano de lançamento. As quatro tuplas recebidas possuem o mesmo tamanho. A função deve retornar uma única tupla que intercala os elementos das tuplas recebidas por argumento.
2. Considere um sistema de posicionamento geográfico que recebe continuamente valores de latitude e longitude, por meio de dois canais de dados independentes, registrados por múltiplos sensores de posição. Os valores de latitude e longitude ficam, cada um, organizados em duas tuplas separadas e de mesmo tamanho, de modo que valores em posições correspondentes de cada tupla compõem uma posição geográfica. O sistema também possui um terceiro canal de dados, que recebe uma string identificando o sensor que realizou uma determinada leitura de posição geográfica. A string identificadora é formada pelo caractere maiúsculo "S" seguido por um caractere numérico. Por exemplo: "S1", "S2", "S3" e assim por diante. As strings identificadoras também estão organizadas em uma terceira tupla, a qual tem o mesmo tamanho das tuplas de latitude e longitude. Posições correspondentes em cada uma das três tuplas armazenam informações pertinentes a uma mesma medição, isto é, a posição geográfica registrada (formada pela informação de latitude e de longitude) e o sensor que realizou esse registro. Por exemplo: para a tupla de latitude (1.1, 2.2), a tupla de longitude (-3.3, 4.4), e a tupla identificadora ('S1', 'S2'), segue que a primeira posição geográfica é formada pelo primeiro elemento da tupla de latitude (que é 1.1), o primeiro elemento da tupla de longitude (que é -3.3), e eles foram registrados pelo sensor identificado no primeiro elemento da tupla de strings (que é 'S1'). A segunda posição geográfica é formada pelo segundo elemento da tupla de latitude (que é 2.2), o segundo elemento da tupla de longitude (que é 4.4), e eles foram registrados pelo sensor identificado no segundo elemento da tupla de strings (que é 'S2').

Sua tarefa é fazer uma função em Python que ajude a agrupar todas as medidas de posição geográfica realizadas por um sensor específico. Para isso, a função deve receber por argumento, na seguinte ordem, a tupla de latitudes (contendo floats), a tupla de longitudes (contendo floats), a tupla de identificadores dos sensores (contendo strings), e uma string que identifica um sensor em particular. A função deve retornar uma tupla de tuplas, em que cada subtupla contém dois elementos, representando, nesta ordem, uma dada medida de latitude e de longitude feitas pelo sensor especificado no quarto de argumento de entrada da função. Por exemplo: para a entrada ((12.2, 13.1, 15.4, 22.3), (-9.9, 11.3, 43.2, 60.5), ('S1', 'S2', 'S3', 'S1'), 'S1'), a saída deve ser ((12.2,

-9.9), (22.3, 60.5)). Ambas as subtuplas correspondem a medições feitas pelo sensor “S1”. Como, na tupla de strings, o sensor “S1” aparece na primeira e na quarta posição, segue que os valores de latitude e longitude nas respectivas posições de cada tupla representam as duas posições geográficas que foram registradas pelo sensor em questão. A primeira subtupla corresponde à primeira posição geográfica registrada pelo sensor “S1”, e ela é formada pelo primeiro elemento da tupla de latitude e pelo primeiro elemento da tupla de longitude. A segunda subtupla corresponde à segunda posição geográfica que foi lida pelo sensor “S1”, e ela é formada pelo quarto elemento da tupla de latitude e pelo quarto elemento da tupla de longitude. Caso não haja medições feitas pelo sensor especificado no quarto argumento da função, a função deve retornar uma tupla vazia. Veja o arquivo de testes para mais exemplos.

3. Escreva uma função em Python que receba três strings de entrada. A função deverá substituir, dentro da string passada como primeiro argumento, todas as ocorrências da string passada como segundo argumento pela string passada como terceiro argumento, e retornar a string resultante após essas substituições. Por exemplo: para a entrada ('abcd', 'ab', 'ef'), a saída deve ser ('efcd'). Se a string passada como segundo argumento não ocorrer na string passada como primeiro argumento, então o programa deve retornar a string do primeiro argumento. Por exemplo: para a entrada ('abcd', 'ef', 'g'), a saída deve ser ('abcd'). O programa também deve ser sensível a maiúsculas e minúsculas. Por exemplo: para a entrada ('Teste', 'te', 'e'), a saída deve ser ('Tese'). Note que o primeiro “Te” não foi substituído por “e”, já que a busca foi por “te” (minúsculo). Veja o arquivo de testes para outros exemplos de funcionamento da função.
4. Para essa questão, você precisará, obrigatoriamente, escrever duas funções. A primeira, que deve terminar com “4a”, deve receber uma tupla contendo valores numéricos (inteiros ou floats) que representam a pontuação de diferentes jogadores em uma rodada de um jogo. Cada posição da tupla representa um jogador. Portanto, o primeiro elemento da tupla representa a pontuação do jogador 1, o segundo elemento da tupla representa a pontuação do jogador 2, e assim por diante. Essa função deve retornar uma tupla contendo inteiros que representam os jogadores que tiveram pontuação abaixo da pontuação média da rodada em questão. Por exemplo: para a tupla de entrada (10,5,8,3,10,6), a pontuação média é 7, e os jogadores que tiveram pontuação abaixo da média foram os jogadores na posição 1, 3 e 5 desta tupla. Sendo assim, a saída, para esta entrada, deve ser a tupla (1,3,5).

A segunda função, que deve terminar com “4b”, deve receber uma tupla de tuplas, representando a pontuação dos jogadores ao longo de múltiplas rodadas. Todas as subtuplas têm o mesmo tamanho e cada uma contém a pontuação dos jogadores em uma rodada específica. Essa função deve retornar uma tupla de tuplas, em que cada subtupla possui dois elementos, nesta ordem: uma string que identifica o jogador, e um inteiro que indica a quantidade de rodadas em que o respectivo jogador ficou com pontuação abaixo da média em questão. A string que identifica o jogador deve, necessariamente, seguir o padrão “JogadorX”, onde “X” representa o identificador do jogador (“Jogador1”, “Jogador2”, ...). Além disso, o jogador que ocupa a posição “0” das subtuplas de pontuação deve ser identificado com o número 1 (pois é o primeiro jogador), o jogador que ocupa a posição “1” das subtuplas de pontuação deve ser identificado com o número 2 (pois é o segundo jogador) e assim por diante.. Por exemplo: para a entrada ((10,5,3), (0,2,8), (6,9,9)), a saída deve ser (('Jogador1', 2), ('Jogador2', 2), ('Jogador3', 1)). O primeiro jogador ('Jogador1') ficou abaixo da média na segunda e terceira rodadas (2 vezes). Logo, a primeira subtupla de saída contém ('Jogador1', 2). O segundo jogador ('Jogador2') ficou abaixo da média na primeira e segunda rodadas (também 2 vezes). Logo, a segunda subtupla de saída contém ('Jogador2', 2). Por fim, o terceiro jogador ('Jogador3') ficou abaixo da média apenas na primeira rodada (1 vez). Logo, a terceira subtupla de saída contém ('Jogador3', 1). **Atenção! Para essa questão (tanto na letra a quanto na letra b), não é permitido o uso de listas, nem em variáveis intermediárias. Utilize apenas tuplas!**