



Quinta Lista de Exercícios

As questões a seguir envolvem o uso das estruturas Dicionários, funções e funções recursivas (explicadas em sala) juntamente com problemas simples e do dia-a-dia, para os quais será necessário o desenvolvimento de scripts em Python. Em cada caso, procure antes resolver de fato o problema para só então começar a escrever código, lembrando sempre que, para cada problema, existem mais de uma solução aplicável.

Dicionários:

1. Escreva um programa em Python que adicione em um dicionário, dados referentes à usuários e senhas de contatos de forma que o nome de usuário seja tratado como chave e a senha como valor.
2. O Python por si só já identifica se uma das chaves está ou não atribuída à algum valor, disparando uma exceção para o usuário. De toda forma, modifique o programa anterior de modo que seja verificada se a chave já tem um valor atribuído e, caso tenha, informe ao usuário, direcionando-o a tentar outra chave.
3. Modifique novamente o programa para que os caracteres da senha sejam trocados pelo caractere “*”, APENAS na exibição.
4. Faça um programa em Python que simule uma lista telefônica e implemente uma forma de pesquisa por número de telefone, que retorne uma mensagem dizendo se o telefone está ou não na lista. Adote como padrão o formato “12345678”, apenas números e sem espaços.
5. A variável atribuída as chaves dos dicionários podem ser de vários tipos, *list*, *int*, *dics* entre outros, escreva um programa em Python que mantenha um dicionário com 3 alunos, onde:
 - a. A matrícula seja tratada com chave no dicionário.
 - b. O primeiro valor da lista, seja o primeiro nome.
 - c. Os segundos, terceiros e quartos valores devem ser notas de 4 bimestres.
6. Em Python, o que garante que as chaves em dicionários sejam únicas em seus usos? Justifique.
7. Altere a lista telefônica para que os contatos possam ter mais de um número cadastrados em seus nomes.
8. Lembra da questão 5? Então, altere o programa para que o nome e as notas do aluno sejam armazenados em um dicionário com as seguintes chaves “nomeAluno”, “nota1”, “nota2”, “nota3” e “nota4”.
9. Implemente um programa em Python que insira 10 pessoas (nome como chave) e suas respectivas idades e sexos (m ou f) em um dicionário, em seguida exiba apenas as pessoas sejam maiores de idade, onde maior idade ≥ 18 anos e depois as pessoas de um mofo geral separadas por sexo.
10. Escreva um programa em Python simule um controle de estoque onde sejam cadastrados 4 produtos, cada um com 10 unidades, o usuário vai poder “tirar” do estoque sempre que quiser, faça com que quando um determinado produto tenha 0 unidade no estique, ele seja eliminado do

dicionário.

Funções:

11. Escreva uma função em Python que receba 2 números inteiros e:
 - a. Imprima a (soma, subtração, divisão e multiplicação) do primeiro pelo segundo.
 - b. Imprima o segundo elevado pelo primeiro.
 - c. Imprima um dicionário onde o primeiro é a chave e o segundo é o valor.
12. Escreva uma função em Python que receba um dicionário de nome(chave):idade(valor) e exiba no formato: "nome" tem "idade" anos de idade.
13. Escreva uma função em Python que receba uma matriz qualquer e informe ao usuário se a mesma é quadrada ou não, exibindo o número de linhas e colunas da mesma.
14. Escreva uma função em Python que receba um dicionário qualquer e exiba em formato de lista os valores contidos no esquema "chave - valor".
15. Escreva uma função em Python que implemente três funções: uma soma e uma de subtração, salve os resultados dessas funções e os mande para uma terceira função, que deve retornar para o usuário a multiplicação dos valores passados.
16. Escreva uma função em Python que receba uma determinada lista de inteiros e retorne uma nova lista apenas com os números ímpares da lista passada como parâmetro.
17. Escreva uma função em Python que imprima para o usuário a tabuada de multiplicação do número que ele passar como parâmetro.
18. Escreva uma função em Python que, assim como a questão anterior, exiba a tabuada de determinado número, a partir do segundo número que ele passar. Ex: tabuada de 5, iniciando a partir do 4.
19. Escreva uma função em Python que mantenha um dicionário, a função deve receber 3 valores, 1) o dicionário em questão 2) o valor a ser inserido e 3) identificador para saber se o valor passar será adicionado ou deletado do mesmo (1 – Adicionar 2 – Deletar).
20. Escreva uma função em Python que exiba todo o fatorial de um determinado número.

Funções recursivas:

21. Escreva uma função em Python que, da mesma forma que a questão anterior, exiba o fatorial de um número.
22. Escreva uma função em Python que simule uma contagem regressiva, onde quando a contagem chegue a zero (0), ele exiba "fim da contagem regressiva." Obs: Para uma simulação mais realista pode-se usar a função *sleep* do Python na biblioteca *time*. (Opcional).
23. Escreva uma função em Python que receba uma lista de inteiros e retorne o valor soma de todos os elementos.
24. Escreva uma função em Python que, modificando o programa anterior, retorne apenas a soma dos valores ímpares da lista.
25. Escreva uma função em Python que exiba o *n*-ésimo número da sequência Fibonacci.
26. Escreva uma função em Python que exiba a soma de todos os números do intervalo de 1 até *n*, onde o *n* deverá ser passado como parâmetro pelo usuário.
27. Escreva uma função em Python que dados dois *n* e *x* parâmetros retorne para o usuário n^x .
28. Escreva uma função em Python que imprima a palavra "recursividade" para sempre.

29. Fale sobre o que pra você, seria a principal diferença entre funções recursivas e iterativas.
30. Na sua opinião, computacionalmente falando, o que se mostra com melhor desempenho, uma função iterativa, ou uma função recursiva? Justifique.