

Lista 3 - Coleções em Python

Resolva os exercícios com a linguagem Python, tente seguir o caminho mais **pythonico** possível.

1. Dada uma lista de palavras, construa um dicionário que mapeia cada palavra para o número de caracteres que ela possui.
2. Dada uma lista de palavras, agrupe-as em um dicionário onde a chave seja uma forma canônica (por exemplo, os caracteres ordenados) e o valor seja a lista de palavras que são anagramas entre si. Essa técnica pode ser utilizada para identificar grupos de palavras com os mesmos caracteres, independentemente da ordem.

Entrada:

```
["amor", "roma", "mora", "carro", "orça", "orca", "arco"]
```

Saída:

```
('a', 'm', 'o', 'r'): ['amor', 'roma', 'mora']  
( 'c', 'o', 'r', 'r'): ['carro']  
( 'a', 'c', 'o', 'r'): ['orça', 'orca', 'arco']
```

3. Dada uma lista de intervalos representados por tuplas (início, fim), escreva um programa que mescle os intervalos que se sobrepõem. Por exemplo, dado os intervalos `[(1, 4), (2, 5), (7, 9)]`, a saída deve ser `[(1, 5), (7, 9)]`.
4. Dada uma lista de números, conte a frequência de cada número e, em seguida, filtre e exiba somente os números que aparecem um número mínimo de vezes (por exemplo, pelo menos 3 vezes).
5. Represente polinômios como dicionários, onde as chaves são os expoentes e os valores são os coeficientes. Por exemplo, o polinômio $3x^2 + 2x + 1$ é representado por `{2: 3, 1: 2, 0: 1}`.

Crie uma função que multiplique dois polinômios e retorne o polinômio resultante, também como dicionário.

6. Dada uma lista de números inteiros e um valor alvo, determine se existe um subconjunto cuja soma seja exatamente igual ao valor alvo.
7. Faça um jogo da Forca utilizando listas. Dada uma palavra, dê algumas chances para o usuário acertar.
8. Faça um programa que leia uma expressão com parênteses. Usando pilhas, verifique se os parênteses foram abertos e fechados na ordem correta. Exemplo:

`(())` **OK** `(())(())` **OK** `(())` Erro

Você pode adicionar elementos à pilha sempre que encontrar abre parênteses e desempilhá-la a cada fecha parênteses. Ao desempilhar, verifique se o topo da pilha é um abre parênteses. Se a expressão estiver correta, sua pilha estará vazia no final.

9. Escreva um programa que compare duas listas. Utilizando operações com conjuntos, imprima:
 - os valores comuns às duas listas
 - os valores que só existem na primeira
 - os valores que existem apenas na segunda
 - uma lista com os elementos não repetidos das duas listas.
 - a primeira lista sem os elementos repetidos na segunda
10. Crie um programa que solicite ao usuário os dados de uma pessoa e armazene essas informações em um dicionário. O dicionário deverá conter as seguintes chaves e seus respectivos valores:
 - **first_name**: Primeiro nome
 - **last_name**: Sobrenome
 - **age**: Idade
 - **city**: Cidade onde a pessoa vive

O programa deve apresentar todas as informações armazenadas no dicionário.

Com uso de listas, armazene dados de várias pessoas. Exiba todas as informações armazenadas.