

Computação I - Python
Departamento de Ciência da Computação
UFRJ
Laboratório 10

1. Escreva uma função que recebe uma lista e retorna uma nova lista sem elementos duplicados. Lembre que os elementos duplicados não precisam aparecer em posições consecutivas. Dica: use um dicionário.
2. Escreva uma função que converte números inteiros entre 1 e 999 para algarismos romanos. Não converta o número para uma string, e use um laço while. Use os três dicionários abaixo:
 - UNIDADES = { 0: "", 1: 'I', 2: 'II', 3: 'III', 4: 'IV', 5: 'V', 6: 'VI', 7: 'VII', 8: 'VIII', 9: 'IX' }
 - DEZENAS = { 0: "", 1: 'X', 2: 'XX', 3: 'XXX', 4: 'XL', 5: 'L', 6: 'LX', 7: 'LXX', 8: 'LXXX', 9: 'XC' }
 - CENTENAS = { 0: "", 1: 'C', 2: 'CC', 3: 'CCC', 4: 'CD', 5: 'D', 6: 'DC', 7: 'DCC', 8: 'DCCC', 9: 'CM' }
3. Construa uma função que receba uma string e retorne um dicionário onde cada palavra dessa string seja uma chave e tenha como valor o número de vezes que a palavra aparece. Por exemplo:

freq_palavras("dinheiro é dinheiro e vice versa")

retorna o dicionário:

{ "dinheiro":2, "é": 1, "e": 1, "vice": 1, "versa":1}

4. Sabe-se que uma molécula de RNA mensageiro é utilizada como base para sintetizar proteínas, no processo denominado de tradução. Cada trinca de bases de RNA mensageiro está relacionado com um aminoácido. Combinando vários aminoácidos, temos uma proteína. Com base na tabela (simplificada) de trincas de RNA abaixo, crie uma função que receba uma string representando uma molécula de RNA mensageiro válida, segundo essa tabela, e retorne a cadeia de aminoácidos que representam a proteína correspondente:

Trinca de RNA	Nome do Aminoácido
UUU	Phe
CUU	Leu
UUA	Leu
AAG	Lisina
UCU	Ser
UAU	Tyr
CAA	Gln

Exemplo: `traducao_rnaM("UUUUUAUCU")` retorna `"Phe-Leu-Ser"`

5. Escreva uma função que recebe uma lista de compras e um dicionário contendo o preço de cada produto disponível em uma determinada loja, e retorna o valor total dos itens da lista que estejam disponíveis nesta loja. Por exemplo, para os dados:

```
lista_de_compras = 'biscoito', 'chocolate', 'farinha'
supermercado = {
    'amaciante':4.99,
    'arroz':10.90,
    'biscoito':1.69,
    'cafe':6.98,
    'chocolate':3.79,
    'farinha':2.99
}
```

O valor retornado pela função será 8.47.

6. Afinidades entre pessoas podem ser descritas através de dicionários, por exemplo se tivermos:

- Leo gosta de Sofia,
- Marcos gosta de Andrea,
- Sofia gosta de Leo,
- Alex gosta de Andrea, e
- Andrea gosta de Marcos

Podemos representar estas relações através do dicionário:

```
afinidades = {
    'Leo': 'Sofia',
    'Marcos': 'Andrea',
    'Sofia': 'Leo',
    'Alex': 'Andrea',
    'Andrea': 'Marcos'
}
```

Escreva uma função que recebe um dicionário de afinidades e retorna uma lista de pares com afinidade mútua, ou seja, onde um gosta do outro. Por exemplo, para o dicionário acima, a função deve retornar:

```
[('Marcos', 'Andrea'), ('Sofia', 'Leo')]
```