**Linguagem de Programação I – Exercícios de Listas**

1. Escreva um programa que recebe uma lista e retorna a lista invertida, para lista abaixo:

L = [ 4, 9, 10, 8, 6] a lista L invertida é igual a [6, 8, 10, 9, 4]

Resolva essa questão sem utilizar a função reverse() para listas

1. Escreva um programa que leia uma string e, em seguida, imprima uma tabela com o **número de ocorrências** de cada caractere na string, sem o uso do método count.
2. Escreva um programa que efetua a **INTERSECÇÃO** entre duas listas, ou seja, os elementos em comum entre as duas listas. Considere que as listas não contêm valores duplicados e não estão ordenadas. Como resultado deve ser gerado uma nova lista e retornada, a nova lista conterá a INTERSECÇÃO das duas listas, exemplos:

A = [7, 2, 5, 8, 4] e B = [4, 2, 9, 5] C = A ∩ B = [2, 5, 4]

A = [3, 9, 11] e B = [2, 6, 1] C = A ∩ B = []

Resolva essa questão sem utilizar a função pronta do Python.

1. Escreva um programa que efetua a **UNIÃO** entre duas listas. Considere que as listas não contêm valores duplicados e não estão ordenadas. Como resultado deve gerada uma terceira lista que conterá a UNIÃO de das listas, exemplo:

A = [7, 2, 5, 8, 4] e B = [4, 2, 9, 5] C = A ∪ B = [7, 2, 5, 8, 4, 9]

A = [3, 9, 11] e B = [2, 6, 1] C = A ∪ B = [3, 9, 11, 2, 6, 1]

Resolva essa questão sem utilizar a função pronta do Python.

1. A **INTERCALAÇÃO** é o processo utilizado para construir uma lista ordenada, de tamanho n+m, a partir de duas listas já ordenadas de tamanhos n e m. Por exemplo, a partir das sequências abaixo:

A=[ 1, 3, 6, 7] e B=[2, 4, 5]

a nova lista C é feita a partir da INTERCALAÇÃO de A e B:

C=[1, 2, 3, 5, 6, 7]

Escreva um programa que efetua a INTERCALAÇÃO entre duas listas, ao final, a sua função devolve a INTERCALAÇÃO das listas em uma única lista como o exemplo acima.