

FUNÇÕES AUXILIARES DO TRABALHO FINAL

Leia atentamente as recomendações a seguir, nela estão alguns dos critérios de correção para as funções auxiliares:

- **Importante!!!** Escolha somente um dos trabalhos para fazer as funções, seja 2048 ou batalha naval, ao enviar as funções, envie num arquivo separado no seguinte formato: `funcoes_auxiliares_nome_aluno.py`
- **MUITO IMPORTANTE!!!** Ao definir o trabalho final, colocar comentários no início do código o trabalho que vai fazer, seu nome e dre. **Exemplo**

```
# Trabalho final - Batalha Naval
# Arquivo de funcoes auxiliares
# Nome do Aluno
# DRE
```

- Documentação da função é essencial
- Não esqueça de colocar os tipos de entrada e saída
- Nomes elucidativos para as variáveis, funções e parâmetros são **MUITO IMPORTANTES**, nos ajudam a entender o seu código e te ajudarão a entendê-lo quando for revê-lo daqui a um mês.
- Não use acentos/cedilha, em nomes de funções variáveis ou parâmetros, o mais simples é sempre melhor. Ao separar nomes compostos recomenda-se o uso de underline (_) ou letras maiúsculas, ex: `area.retangulo`, `areaRetangulo`
- Tente escrever o código o mais simples possível, se houver alguma parte que seja de difícil compreensão, recomenda-se o uso de comentários ex: (`# essa linha de código faz tal coisa`), mas use-o com moderação, muitos comentários dificultam a compreensão do código.
- Sua solução não deve usar de conceitos e bibliotecas que não foram vistos ao longo das últimas semanas.
- **para fazer a entrega desta atividade prática, escreva suas funções no editor do IDLE, salvando todas em um único arquivo.**
- **as questões que tiverem matéria que ainda não foi apresentada, serão desconsideradas**
- **Trabalhos plagiados/copiados/iguais serão zerados**

1. 2048

- (1) Escreva uma função que recebe uma lista numérica e some os números adjacentes que sejam iguais, a soma deve ser feita da esquerda para a direita. A função deve modificar a própria lista, retornando True se houve a modificação.

Dica: A questão Repetidos do MT07 pode ajudar nesse problema **Cuidado com o exemplo abaixo:**

```
>> l = [1,2,2,4,8]
>> funcao_auxiliar1(l) #nome generico nao use esse nome
>> l
[1,0,4,4,8] # l nao pode ser [1,0,0,0,16]
```

- (2) Modifique ligeiramente a função 1, para receber dois parâmetros, a lista e a direção (esquerda/direita) para onde deseja-se realizar a soma

Exemplos:

```
>> l = [1,2,2,4,8]
>> funcao_auxiliar1(l,'dir') #nome generico nao use esse nome
>> l
[1,0,4,4,8]
>> l2 = [1,2,2,4,8]
>> funcao_auxiliar1(l2,'esq')
>> l2
[1,4,0,4,8]
```

- (3) Crie uma função que dado uma lista numérica de tamanho qualquer, junte todos os elementos diferentes de zero no extremo direito da lista.

Exemplo:

```
>> l = [1,2,2,4,0]
>> funcao_auxiliar2(l) #nome generico nao use esse nome
>> l
[0,1,2,2,4]
```

- (4) Modifique a função acima de maneira que ela receba mais um parâmetro, a extremidade que se deseja juntar.

Exemplo:

```
>> l = [0,1,2,2,4,0]
>> funcao_auxiliar2(l, 'dir') #nome generico nao use esse nome
[0,0,1,2,2,4]
>> funcao_auxiliar2(l, 'esq')
[1,2,2,4,0,0]
```

Essa função não precisa modificar a lista original, pode devolver uma outra modificada, mas fiquem livres decidir

2. BATALHA NAVAL

- (1) Escreva uma função que recebe uma lista numérica de cujos itens são únicos, ou seja, não pode existir uma lista dessa maneira : `[1,1,2,2,3,4,5]`, pois tem dois 1 e dois 2, e uma lista cujos itens desejamos retirar da lista original. A função deve excluir esses elementos da lista original. Retorna True caso a operação seja possível

Exemplo:

```
>> l = [0,1,2,4,7,8]
>> ret = [1,2,4]
>> funcao_auxiliar1(l, ret) #nome generico nao use esse nome
>> l
[0,7,8]
```

- (2) Escreva uma função que recebe um número e devolve uma lista de listas de um único elemento que vai do 0 até o número-1.

Exemplo:

```
>> funcao_auxiliar2(10) #nome generico nao use esse nome
[[0],[1],[2],[3],[4],[5],[6],[7],[8],[9]]
>> funcao_auxiliar2(5)
[[0],[1],[2],[3],[4]]
```
