

Lista de Exercícios 3: *entrega → 25/04 (entregar soluções em PAPEL, pode fazer em grupo)*

- **Obs1:** Os exercícios envolvem toda a matéria da VAE1, desde a aula01 até a aula09.
- **Obs2:** As soluções para os exercícios 5 a 9 devem funcionar para qualquer lista (independente do tamanho ou conteúdo) e não apenas para as listas apresentados nos exemplos.

(1) Faça um programa que calcule e imprima a **soma dos 50 primeiros termos** da série S, cujos 4 primeiros termos são dados abaixo:

$$S = (1/2) - (3/4) + (5/8) - (7/16) + \dots$$

(2) Criar um programa que leia a idade e a escolaridade (S = Nível Superior e N = Não Possui Nível Superior) de várias pessoas. Calcule e imprima a idade média, o total de pessoas com nível superior e o total de pessoas que não possuem o nível superior. A entrada de dados deve encerrar quando se digita – 1 para a idade (flag de saída). Ignore um dado de entrada se a escolaridade digitada for diferente de ‘S’ ou ‘N’ ou se a idade digitada for menor que 0.

(3) Faça um programa que tabule a seguinte função:

$$g(x,y) = \begin{cases} \frac{x^2 - y^2}{x^2 + y^2}, & \text{se } (x,y) \neq (0,0) \\ 0 & \text{se } (x,y) = (0,0) \end{cases}$$

Para $x = 0, 1, 2, \dots, 100$;

e $y = 0, 1, 2, \dots, 5$ para cada valor de x .

(4) Crie um programa que receba como entrada um número inteiro x e imprima como saída o número de dígitos de x usando uma solução que **não converta x para string**. Exemplos:

- $9999 \rightarrow 4$
- $1048576 \rightarrow 7$

(5) Faça um programa que multiplique todos os elementos de uma lista de reais u de tamanho n .

Por exemplo, se $u = [10, 0.20, 4] \rightarrow 8.0$

(6) Faça um programa capaz de verificar se todos os elementos de uma lista de strings w de tamanho n são iguais. A saída do seu programa, deve ser da seguinte forma:

- Caso todos os elementos sejam iguais, imprimir a frase “Todos são iguais”.
- Caso contrário, imprimir a frase “Nem todos são iguais”.

Ex.: se $w = ['ENCE', 'ENCE', 'ENCE'] \rightarrow$ “Todos são iguais”
se $w = ['ENCE', 'UFF', 'ENCE', 'ENCE'] \rightarrow$ “Nem todos são iguais”

(7) Faça um programa capaz de verificar se o primeiro e o último elemento de uma lista de strings w de tamanho n são iguais e todos os demais são diferentes. Se isso ocorrer, o programa deve imprimir “Sim”, caso contrário “Não”. Veja o exemplo:

Se: $w = ['ENCE', 'UFF', 'ENCE', 'ENCE'] \rightarrow \text{“Não”}$

Se: $w = ['ENCE', 'UFF', 'UERJ', 'ENCE'] \rightarrow \text{“Sim”}$

(8) Faça um programa que crie uma nova lista b através da concatenação dos elementos de uma lista a com uma faixa de inteiros de 1 a k . Veja o exemplo:

Para: $a = ['oi', 'tchau']$ e $k = 3 \rightarrow b = ['oi1', 'tchau1', 'oi2', 'tchau2', 'oi3', 'tchau3']$

(9) Faça um programa que diga qual é o 2º menor valor de uma lista u contendo n números, **sem utilizar método sort()**. Exemplo:

Para: $u = [30, 50, 50, 35, 30, 40, 30, 100] \rightarrow$ o segundo menor valor é 35

(10) (Crivo de Eratóstenes) Um **número primo** é um número natural maior que 1 que não pode ser formado pela multiplicação de outros dois naturais menores. O “Crivo de Eratóstenes” é um método diferente, porém muito prático de se encontrar números primos até um certo valor limite. Esse método deverá ser utilizado nesta questão, para que sejam determinados todos os números primos menores do que 1000. Para isto, escreva um programa composto pelos **passos** descritos a seguir:

(i) Crie uma lista de inteiros com 1001 posições (de 0 a 1000). Depois atribua o valor 1 para todas as posições desta lista, **exceto** as posições 0 e a posição 1. Ambas deverão receber o valor 0.

0	0	1	1	1	1	1	1	1	1	1	1	...	1	1
0	1	2	3	4	5	6	7	8	9	10	11	...	999	1000

(ii) Em seguida, **começando da posição 2 da lista**, sempre que for encontrado um elemento com valor igual a 1, faça um laço pelo restante da lista e defina como zero todos os elementos **cujo subscrito seja múltiplo** do subscrito com valor 1.

Exemplo: para o subscrito 2 da lista, todos os elementos posteriores à posição 2 que estiverem armazenados em subscritos múltiplos de 2 serão definidos como zero (subscritos 4, 6, 8, 10 etc.). Idem para o subscrito 3 da lista (subscritos 6, 9, 12, 15, 18 etc.). E assim, para os demais subscritos.

Ao final do processamento, os elementos da lista com subscritos primos, permanecerão 1. Todos os outros elementos estarão com o valor 0, conforme ilustra a figura abaixo.

0	0	1	1	0	1	0	1	0	0	0	1	...	0	0
0	1	2	3	4	5	6	7	8	9	10	11	...	999	1000

Para finalizar o programa, crie uma rotina que percorra a lista e imprima na tela todos números primos entre 1 e 1000 (basta **imprimir** todos **os subscritos** dos elementos que contêm o valor 1).