
Quinta Lista de Exercícios

As questões a seguir envolvem o uso das estruturas de lista, listas de listas e matrizes, juntamente com questões lógicas e do dia-a-dia, para os quais será necessário o desenvolvimento de *scripts* em linguagem Python. Em cada caso, procure antes, de fato, desenvolver uma solução para o problema, para só então escrever código, lembrando que, cada problema pode ser resolvido de muitas maneiras diferentes.

1. Escreva um programa em Python que componha uma lista que seja capaz de receber 10 valores, desses, apenas número positivos.
2. Escreva um programa em Python que ordene a lista preenchida na questão anterior. Detalhe, pesquisar sobre o método bolha e como ele funciona.
3. Escreva um programa em Python que simule um jogo de dados, gerando números aleatórios dentro do intervalo 1 - 6, jogue esse "dado" 10 vezes, salve os números que foram sorteados e informe ao usuário qual o maior, o menor e as posições desses números dentro da lista. Detalhe, com e SEM o uso das funções `max()` e `min()`.
4. Dada a lista `[9,1,3,2,5,2,5,5,7,0,4,1,6,5,9]` crie um programa em Python que:
 - a. Some todos o número da esquerda para a direita.
 - b. Subtraia todos os números da direita para a esquerda (Ex: $9-5=4 \rightarrow 4-6= -2 \dots$).
 - c. Some a soma dos 3 maiores números com a soma dos três menores.
5. Escreva um programa em Python que, dada uma determinada palavra, verifique se ela é ou não um palíndromo. Uma palavra é dita um palíndromo de ela se mantiver igual quando percorrida pelos dois lados, gerando o mesmo resultado. Ex: ama, arara, aba e etc.
6. Veja o código a seguir e preencha os blocos de código (funcionalidade comentada) e por fim exiba as listas no fim da execução. Obs.: os condicionais dos `ifs` podem ser criados a escolha do aluno.

```
1. candidatos = [22,45,78,5,45,63,99,2,51,8]
2. aprovados = []
3. em_espera = []
4. if [bloco de código 1]:
5.     print("aprovado")
6.     [bloco de código 2] #insere na lista de aprovados
7. elif [bloco de código 3]:
8.     print ("lista de espera")
9.     [bloco de código 4] #insere na lista de espera
10. else:
11.     print ("quem não estiver nas listas, foi reprovado")
print("FIM!")
```

7. Escreva um programa em Python que armazene 10 inteiros positivos diferentes, obtenha a média desses números e exiba todos os números abaixo dessa média.
8. Escreva um programa em Python que mantenha duas listas, uma para contatos e outra para seus respectivos números, insira 5 contatos nessa lista e exiba ela no fim da execução, detalhe, o número precisa ter somente 8 dígitos, não mais e nem menos.
9. O que o seguinte código faz?

```

1. frase = list ("o rato roeu a roupa do rei de roma")
2. print ("frase:", frase)
4. for x in range(len(frase)):
6. if (frase[x] == ' '):
7.     frase[x] = '-'
8. print ("frase:", frase)

```

10. Escreva um programa em Python que preencha uma lista com uma palavra ou qualquer sequência de caracteres e conte para o usuário quantas vogais e quantas consoantes existem ali.
11. Implemente uma lista de listas em Python que, em uma posição guarde nomes femininos em uma lista e nomes masculinos em outra, até que a palavra “sair” seja digitada pelo usuário.
12. Crie uma matriz n por n em Python e: 1) preencha a mesma com números aleatórios 2) exiba a soma da diagonal principal com a secundária e 3) o maior e menor elemento da matriz.
13. Crie mais uma matriz quadrada q , dessa vez completando-a inicialmente com apenas um valor n , verificando se o q (colunas e linhas) da matriz QUADRADA é ímpar ou par, faça:
 - a. Se ímpar, preencha o valor central da matriz com um elemento diferente do restante.
 - b. Se par, preencha os quatro valores centrais da matriz com um elemento diferente do restante.

Matriz ímpar:					Matriz par:					
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1	1	0	0
0	0	0	0	0	0	0	1	1	0	0
0	0	0	0	0	0	0	0	0	0	0
					0	0	0	0	0	0

14. Crie um programa em Python que data uma lista preenchida aleatoriamente, retorne para o usuário os múltiplos de um número n passado pelo usuário.
15. Crie um programa em Python que simule um To-Do List, onde cada tarefa a ser feita seja inserida em uma lista no padrão **identificador, descrição, status** (1 para feito e 0 para pendente.), o modelo a ser usado deve ser EXCLUSIVAMENTE uma lista de listas.
16. Escreva um programa em Python que simule uma urna eletrônica (lembra da outra

lista? Esquece aquele esquema) em que sejam cadastrados n candidatos, o programa deve computar quantos votos cada candidato recebeu em p turnos e exibir a média de cada candidato (soma de votos / turnos) ao final da execução,

17. Analise, sem a ajuda de um compilador, quais dos comandos a seguir dão ou não certo, e a causa, nos casos em que não dá.

- a. `list1 = [0] * tamanho`
- b. `list2 = [[], [], []]`
- c. `list3 = [0, "nome", 'j']`
- d. `list4 = ["nome", 'nome', 12.5]`
- e. `list5 = [list1, list2, list3, [list4, [list5]]]`

18. Escreva um programa em Python que compare dados de taxa de download de 5 provedores de internet e ao final da execução, informe para o usuário, o provedor com melhor taxa.

19. Com base na questão anterior, modifique o programa para que seja possível: 1) Inserir quantos provedores sejam necessários. 2) seja possível capturar a menor taxa e 3) excluir da lista o provedor com menor taxa.

20. Cite quais funções e dados foram usadas em cada estado que a seguinte lista assume.

- a. `list = [0,1,2,3,4,5,6,7,8,9]`
- b. `list = [1,3,4,5,6,7]`
- c. `list = [1,2,3,4,7,8,9,3]`
- d. `list = [1,2,3,3,4,7,8,9]`
- e. `list = [9,8,7,4,3,3,2,1,1,2,3,3,4,7,8,9]`
- f. `list = [1,1,2,2,3,3,3,3,4,4,7,7,8,8,9,9]`

21. Escreva um programa em Python que teste n números aleatórios e mantenha duas listas, a dos números pares e ímpares, e exiba a lista com uma estrutura de repetição

22. Escreva um programa em Python que simule um dado, jogue esse dado n vezes e exiba, ao final da execução, quantas vezes, cada número foi sorteado.

23. Escreva um programa em Python que simule uma matriz $m \times n$ onde seu preenchimento se dê através de uma sequência iniciada a partir de um número inicial aleatório.

24. Depois de todas as questões feitas, defina o que são listas e cite pelo menos 3 benefícios das mesmas.

25. Para saber quanto um número binário vale, se visto em decimal, geralmente se usa uma contagem de 1 até 128, considerando apenas os 1s, de forma que o valor é sempre somado da direita para a esquerda. Ex.: `11101b == 29d`, tal que, da direita para a esquerda temos: `1+4+8+16 == 29`. Visto isso, escreva um programa em Python em que a parte da entrada seja binária e, ao fim da execução, seja retornado para o usuário, a sua forma decimal. Obs.: Sem usar a função `bin()` do Python.

26. Dada a questão anterior, gere 10 números binários, converta-os e exiba o maior deles (em decimal e em binário) ao fim da execução.

27. Escreva um programa em Python que tenha uma lista de valores aleatórios entre 100 e 1000, percorra essa lista e multiplique todos os valores por 0.2, caso o resultado seja maior de 100, insira o valor multiplicado em uma nova lista, ao fim, exiba a média dos valores que passaram na verificação da multiplicação juntamente com o maior valor da lista.
28. Simule três partidas de “ímpar ou par” entre dois usuários e guarde os pontos de quem ganhar em duas listas, no fim, exiba quem ganhou.
29. Qual sequência de funções é necessária para que se elimine o maior elemento de uma lista gerada aleatoriamente, de modo que o índice do maior elemento seja 0.
30. (Desafio) Implemente em uma matriz 3x3 as 8 verificações possíveis em um jogo da velha.