

Capítulo 5

Listas

1. Escreva a função de dois argumentos, `lista_codigos`, que recebe uma cadeia de caracteres e que devolve a lista contendo os códigos “Unicode” de cada um dos caracteres da lista. Por exemplo:

```
>>> lista_codigos('bom dia')
[98, 111, 109, 32, 100, 105, 97]
```

2. Escreva a função de dois argumentos, `remove_multiplos`, que recebe uma lista de inteiros e um inteiro e que devolve a lista que resulta de remover todos os múltiplos do segundo argumento da lista original. Por exemplo:

```
>>> remove_multiplos([2, 3, 5, 9, 12, 33, 34, 45], 3)
[2, 5, 34]
```

3. Escreva a função, `soma_cumulativa`, que recebe uma lista de números e que devolve uma lista que contém a soma cumulativa da lista recebida, ou seja, o elemento na posição i da lista devolvida contém a soma de todos os elementos da lista original nas posições de 0 a i . Não é necessário validar os dados de entrada. Por exemplo,

```
>>> soma_cumulativa([1, 2, 3, 4, 5])
[1, 3, 6, 10, 15]
```

4. Uma *matriz* é uma tabela bidimensional em que os seus elementos são referenciados pela linha e pela coluna em que se encontram. Uma matriz pode ser representada como uma lista cujos elementos são listas, cada uma destas sub-listas representa uma linha. Com base nesta representação, escreva a função `elemento_matriz` que recebe três argumentos, uma matriz, uma linha e uma coluna e que devolve o elemento da matriz que se encontra na linha e coluna indicadas. A sua função deve permitir a seguinte interação:

```
>>> m = [[1, 2, 3], [4, 5, 6]]
>>> elemento_matriz(m, 0, 0)
1
>>> elemento_matriz(m, 0, 3)
ValueError: elemento_matriz: indice invalido, coluna 3
```

5. Considere uma matriz como definida no exercício anterior. Escreva uma função em Python que recebe uma matriz e que a escreve sob a forma

$$\begin{array}{cccc} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{array}$$

6. Considere o conceito de matriz. Escreva a função `soma_mat` que recebe como argumentos duas matrizes e devolve uma matriz correspondente à soma das matrizes que são seus argumentos. Sendo a e b as matrizes a somar, os elementos da matriz produto são dados por

$$s_{ij} = a_{ij} + b_{ij}$$

Por exemplo:

```
m1 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
m2 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
escreve_matriz(soma_mat(m1, m2))
2 4 6
8 10 12
14 16 18
```

7. Considere o conceito de matriz. Escreva a função `multiplica_mat` que recebe como argumentos duas matrizes e devolve uma matriz correspondente ao produto das matrizes que são seus argumentos. Sendo a e b as matrizes a multiplicar, os elementos da matriz produto são dados por

$$p_{ij} = \sum_{k=1}^n a_{ik}b_{kj}$$

8. A sequência de Racamán,

$$0, 1, 3, 6, 2, 7, 13, 20, 12, 21, 11, 22, 10, 23, 9, 24, \dots$$

é uma sequência de números inteiros não negativos, definida do seguinte modo: (1) o primeiro termo da sequência é zero; (2) para calcular o n -ésimo termo, verifica-se se o termo anterior é maior do que n e se o resultado de subtrair n ao termo anterior ainda não apareceu na sequência, neste caso o n -ésimo termo é dado pela subtração entre o $(n-1)$ -ésimo termo e

n ; em caso contrário o n -ésimo termo é dado pela soma do $(n - 1)$ -ésimo termo com n . Ou seja,

$$r(n) = \begin{cases} 0 & \text{se } n = 0 \\ r(n-1) - n & \text{se } r(n-1) > n \wedge (r(n-1) - n) \notin \{r(i) : i < n\} \\ r(n-1) + n & \text{em caso contrário} \end{cases}$$

Escreva uma função que recebe um inteiro positivo, n , e devolve uma lista contendo os n primeiros elementos da sequência de Racamán. Por exemplo:

```
>>> seq_racaman(15)
[0, 1, 3, 6, 2, 7, 13, 20, 12, 21, 11, 22, 10, 23, 9]
```

9. Escreva a função `numero_occ_lista`, que recebe uma lista e um número, e devolve o número de vezes que o número ocorre na lista e nas suas sublistas, se existirem. Não é necessário validar os argumentos. Por exemplo,

```
>>> num_occ_lista([1, 2, 3, 4, 3], 3)
2
>>> num_occ_lista([1, [[1]], 2, [[2]], 2], 2)
3
```

10. Uma chave do euromilhões é constituída por cinco inteiros ordenados, entre 1 e 50 e sem repetições e por dois números diferentes, também ordenados, entre 1 e 12. Escreva uma função sem argumentos que devolve aleatoriamente uma lista contendo duas listas, cada uma delas contendo os constituintes de uma chave do euromilhões. Para a geração de números aleatórios utilize a função `random()`, existente na biblioteca `random`, que devolve aleatoriamente um real no intervalo $[0, 1[$.