

1. Determine os valores das seguintes expressões

- a) $4/3$ b) $4\%3$ c) $2*3**2+1$ d) $\text{len}(\text{range}(999))$
 e) $[j+1 \text{ for } i \text{ in } \text{range}(3) \text{ for } j \text{ in } \text{range}(2) \text{ if } i+j<3]$

2. Escreva uma função `roda(s)` que retorna a string que resulta de `s` por “rotação” das letras minúsculas de 1 posição para a frente:

a -> b -> c -> ... -> y -> z -> a

Por exemplo `roda("As Azas!")` -> `"tabt"`. Note que todos os caracteres que não são letras minúsculas são eliminados.

3. Escreva uma função `num(b, lista)` que recebe como argumentos

- um inteiro `b` ≥ 2
- uma `lista` de inteiros compreendidos entre 0 e `b-1`

e retorna o inteiro cujos dígitos da representação na base `b` são, por ordem, os inteiros existentes em `lista`. Exemplos:

`num(10, [5,1,2,9])` -> 5129
`num(2, [1,1,0,0])` -> 12

Sugestão. Processe os dígitos de `lista` da esquerda para a direita.

4. Escreva uma função `p(lista)` que tem como argumento uma lista `lista` e retorna a sua “profundidade”, assim definida:

- se `lista` não contém elementos que sejam listas: `p(lista)=1` (ex: `p([4,2,6])` -> 1)
- se `lista` contém como elementos listas l_1, l_2, \dots, l_k :

$$p(\text{lista}) = 1 + \max\{p(l_1), p(l_2), \dots, p(l_k)\}$$

Ex: `p([4, [2], [2, [3]], 5])`:

como `p([2])` -> 1
 e `p([3])` -> 1
 temos `p([2, [3]])` -> 2
 logo `p([4, [2], [2, [3]], 5])` =
 $1 + \max\{1, 2\} = 1+2 = 3$

Sugere-se que inclua o caso: `p(li)=0` quando `li` não é lista; exemplo: `p(5)=0`.

5. Escreva uma função $f(l_i)$ que tem como argumento uma lista l_i cujos elementos são inteiros ou listas de inteiros (como no problema anterior) e retorna uma lista dos inteiros existentes em l_i , conforme se exemplifica:

$$f([4, 5, [2], [3, [4]], [[]]]) \rightarrow [4, 5, 2, 3, 4]$$

Sugestão: na construção da lista que é retornada, utilize:

- Recursão (a função que se está a definir chama-se a si mesma, como no problema anterior).
- O operador “+” (concatenação de listas).