

# Programação I

## Funções, tipos e condicionais

*mais exercícios*

**Sumário:** *Funções, tipos de dados e condicionais.*

1. Implemente a função *sucessor()* que, dado um número inteiro, devolve o seu sucessor.
2. Implemente a função *quadradoDoSucessor()* que, dado um número inteiro, devolve o quadrado de  $n + 1$ . Utilize a função definida no exercício anterior.
3. Implemente a função para determinar a velocidade de um carro, sabendo que a aceleração  $a$  é constante e decorreram  $t$  segundos desde que foi registada a velocidade inicial  $v_0$ .

$$v = v_0 + a * t$$

4. Implemente a função para determinar a posição  $p$  de um carro,  $t$  segundos após ter saído da posição  $p_0$  com velocidade inicial  $v_0$  e aceleração  $a$ .

$$p = p_0 + v_0 * t + \frac{1}{2} * a * t^2$$

5. A função *len()* devolve o número de caracteres de uma string; por exemplo *len('Hello')* devolve o número 5. Escreva uma função chamada *right\_justify()*, que recebe uma string  $s$  como argumento, e escreve a string com espaços suficientes à esquerda, de modo a que o último carácter da string fique na coluna 70 da consola.

```
>>>right_justify('Programação I')
>>>
```

Programação I

6. Implemente a função que calcula os custos de envio de uma encomenda de livros. Esta função *custos\_envio()* tem 3 parâmetros: o nº de livros, o custo para a primeira cópia, o custo unitário para as seguintes.
7. Utilizando a função anterior, implemente a função *custo\_encomenda()* que calcula o custo de uma encomenda de livros. A função tem como argumento, o nº de cópias encomendadas e o preço unitário do livro.
8. Implemente a função *tempo\_decorrido()* que calcula o tempo necessário para correr uma distância  $d$  a uma velocidade constante  $v$ ; a distância é indicada em  $km$  e a velocidade em  $km/h$ .
9. Utilizando a função do exercício anterior, implemente a função *hora\_chegada()* calcula a hora de chegada de uma corrida que tem início à hora  $h$ . Esta corrida tem 3 troços de dimensão distinta:
  1. troço 1: 2 km, velocidade média de  $7.5km/h$
  2. troço 2: 6 km, velocidade média de  $10.9km/h$
  3. troço 3: 2 km, velocidade média de  $7.5km/h$
10. Implemente a função *triangulo()* que recebe três valores. A função deverá verificar se os valores formam um triângulo e, em caso afirmativo, indicar qual tipo de triângulo.

```
>>> triangulo(3,9,6.5)
>>> Os valores inseridos 3, 9, 6.5 formam um triângulo escaleno
```

**Dicas:**

- existe um triângulo quando a soma de quaisquer dois lados for maior que o terceiro;
- um triângulo equilátero tem três lados iguais; um triângulo isósceles tem dois lados iguais e um triângulo escaleno tem os três lados diferentes.