

Programação I

Funções (ficha 6)

1. Implemente a função `sucessor()` que, dado um número inteiro n , devolve o seu sucessor $n + 1$.
2. Implemente a função `quadradoDoSucessor()` que, dado um número inteiro n , 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. Implemente a função que calcula os custos de envio de uma encomenda de livros. Esta função `custosEnvio()` tem 3 parâmetros: o nº de livros n , o custo de envio para a primeira cópia c , o custo de envio unitário para as seguintes cs .

$$custo_envio = c + cs * (n - 1)$$

6. Utilizando a função anterior, implemente a função `custoEncomenda()` que calcula o custo de uma encomenda de livros. A função tem como argumentos, o nº de cópias encomendadas e o preço unitário do livro, e o peso de cada livro em kg. O custo de envio de uma cópia é de 3 Eur, cópias seguintes acrescem o custo de 0.2Eur por kg.
7. Implemente a função `tempoDecorrido()` 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 .
8. Utilizando a função do exercício anterior, implemente a função `horaChegada()` que devolve a hora de chegada (como real) de uma corrida que tem início à hora `hp` e minuto `mp`. Assuma que não existe mudança de dia. 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 $7.5km/h$
9. Implemente a função recebe 3 valores (float), e indica se estamos em presença de um triângulo ou não: existe um triângulo quando a soma de quaisquer dois lados for maior que o terceiro.
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. A função devolve: -1 quando não há triângulo; 1 quando o triângulo é escaleno (três lados diferentes); 2 quando o triângulo é isósceles (dois lados diferentes); 3 quando o triângulo é equilátero (três lados iguais)