

# Aula prática #7 – Funções e Apontadores

## Problema 1

---

Escreva uma função que recebe dois valores inteiros por referência e devolve por retorno o endereço do valor maior. Escreva um programa para testar a sua função. Considere o seguinte protótipo para a função:

```
1  int *vmaior(int *valor1, int *valor2);
```

### Exemplo

```
1  Insira dois valores: 56 32
2  Endereco do maior: 0x0016F838
3  Valor: 56
```

## Problema 2

---

Escreva uma função chamada **ordena** que recebe três valores e os ordena por ordem crescente. Teste a sua função com um programa que pede ao utilizador três números, invoca a função **ordena** e depois imprime o resultado no ecrã. Considere o seguinte protótipo para a função:

```
1  void ordena(int *valorA, int *valorB, int *valorC);
```

### Exemplo

```
1  Insira os valores a ordenar: 43 65 17
2  Valores a, b, c ordenados por ordem crescente: 17 43 65
```

## Problema 3

Escreva uma função chamada **horasMin** que converte um valor em minutos para horas:minutos. A sua função deve ainda retornar 1 se o número de horas:minutos for superior a um dia e 0 caso contrário. Para testar a sua função escreva um programa que usa a função **horasMin** para efetuar a conversão e que depois imprime o resultado no ecrã.

```
1  int hoursMin(int totalMins, int *hours, int *minutes);
```

### Exemplo

```
1  Insira o total de minutos:568
2  568 minutos correspondem a 09h:28m e nao e' superior a 1 dia.
3
4  Insira o total de minutos:4689
5  4689 minutos correspondem a 78h:09m e e' superior a 1 dia.
```

## Problema 4

Implemente duas funções que convertam coordenadas cartesianas em coordenadas polares e vice-versa. Deverá definir as seguintes funções:

```
1  void cartesianas_polares(float x, float y, float *r, float *theta);
2  void polares_cartesianas(float r, float theta, float *x, float *y);
```

As relações entre coordenadas cartesianas em coordenadas polares são definidas por:

$$x = r.\cos\theta$$

$$y = r.\sen\theta$$

$$r = \sqrt{x^2 + y^2}$$

$$\theta = \text{atan2}(y, x)$$

Escreva um programa que lhe permita testar as funções que desenvolveu.

## Problema 5

---

Implemente um procedimento que calcule o quociente e o resto da divisão inteira de dois números inteiros, sem utilizar os operadores % e /. Utilizando o procedimento que definir, implemente uma função que determina a soma dos dígitos de um número inteiro e outra função que teste se um número é par. Deverá definir as seguintes funções:

```
1 void quociresto(int dividendo, int divisor, int *quociente, int *resto);  
2 int soma(int n);  
3 int par(int n);
```

Escreva um programa que lhe permita testar as funções que desenvolveu.