

Ficha de Trabalho N.º 5

Objectivos: Funções e *arrays*.

1 - Escreva as seguintes funções sobre o tipo `char`:

	Função	Devolve
a)	<code>int isdigit(char c)</code>	Verdade quando <code>c</code> é um dígito e Falso c.c.
b)	<code>int isalpha(char c)</code>	Verdade quando <code>c</code> é uma letra e Falso c.c.
c)	<code>int isalnum(char c)</code>	Verdade quando <code>c</code> é um carácter alfanumérico e Falso c.c.
d)	<code>char tolower(char c)</code>	Devolve <code>c</code> transformado na minúscula correspondente
e)	<code>char touooer(char c)</code>	Devolve <code>c</code> transformado na maiúscula correspondente

Nota: Obtém-se acesso a estas funções através da directiva

`#include <ctype.h> // Funções sobre o tipo char (ctype -> char type)`

2 - Escreva as funções a seguir indicadas de modo que devolvam os resultados descritos:

	Função	Devolve
a)	<code>int resto (int a, int b)</code>	O resto da divisão de <code>a</code> por <code>b</code>
b)	<code>int impar (int x)</code>	Verdade se <code>x</code> for ímpar e Falso c.c.
c)	<code>int perfeito (int n)</code>	Verdade se <code>n</code> for “perfeito” (igual à soma dos divisores de <code>n</code> , inferiores a <code>n</code>) e Falso c.c.
d)	<code>int primo (int n)</code>	Verdade se <code>n</code> for “primo” (apenas divisível por 1 e por <code>n</code>) e Falso c.c.

3 - No século I d.C. os números naturais dividiam-se em três categorias:

REDUZIDOS: os superiores à soma dos seus divisores;

ABUNDANTES: os inferiores à soma dos seus divisores;

PERFEITOS: os que são iguais à soma dos seus divisores.

NOTA: Nesta definição exclui-se o próprio número do conjunto dos seus divisores.

Escreva uma função que liste os inteiros entre `a` e `b`, $a < b$, classificando-os de acordo com esse critério, e que escreva também o total de cada uma das categorias.

4 - Elabore um programa que determine o cubo de um número inteiro *n*. O número *n* deve ser pedido ao utilizador através de uma função e o seu cubo deve ser calculado através de outra função.

- 5 - Escreva uma função que determine o maior de dois números dados. Teste a função num pequeno programa.
- 6 - Elabore um programa que:
- Leia as n componentes de um vector;
 - Escreva as n componentes de um vector;
 - Determine a posição em que se encontra a maior componente.
- Cada uma destas tarefas deve ser realizada por uma função.
- 7 - Escreva uma função que determine o produto interno de dois vectores de n componentes inteiras.
- 8 - Dado um vector x com n componentes inteiras, escreva funções que permitam realizar as seguintes operações:
- Efectuar a permutação circular do vector dado;
 - Trocar a componente da posição p com a da posição q .
- 9 - Considere uma matriz quadrada com $n \times n$ elementos inteiros. Elabore um programa que lhe permita:
- Ler os $n \times n$ elementos da matriz;
 - Mostrar no monitor os $n \times n$ elementos da matriz;
 - Determinar o valor mínimo da matriz;
 - Verificar se a matriz é ou não simétrica;
 - Determinar a transposta da matriz;
 - Calcular a soma de duas matrizes dadas.
- Cada tarefa deve ser realizada por uma função.