



12/01/2007

Exame da Época Normal

Duração: 2h 30m

Atenção:

- ✓ Para cada um dos problemas propostos no exame deve apresentar:
 - a análise do problema, especificando os *dados de entrada*, os *resultados pretendidos* e as *acções a efectuar*
 - o respectivo algoritmo em pseudocódigo
 - a implementação do programa em linguagem C.
- ✓ As funções desenvolvidas devem ser portáveis.
- ✓ A prova é sem consulta.
- ✓ Faça uma pergunta em cada folha de prova e não escreva na 1ª página de cada uma das folhas.

1. (5.0 valores) Desenvolva um programa que leia uma sequência de caracteres terminada por '.'. À medida que os caracteres vão sendo lidos é escrita no écran, para cada um dos caracteres, a seguinte mensagem "Letra", "Dígito" ou "Outro", consoante se trate de uma letra, de um dígito ou de outro carácter, respectivamente. Após terminada a leitura dos caracteres, o programa deve informar o utilizador sobre o número de caracteres existentes na sequência lida, excluindo-se o '.' final.

Exemplo de execução do programa:

```
Escreva um caracter: 6
Dígito

Escreva um caracter: P
Letra

Escreva um caracter: !
Outro

Escreva um caracter: a
Letra

Escreva um caracter: .
Foram lidos 4 caracteres
```

Nota: Esta questão deve ser resolvida sem utilizar "arrays". Resoluções que façam uso destas estruturas de dados, serão cotadas com zero valores.

2.

- a) (4.5 valores) Faça uma **função** que receba como argumentos um vector de inteiros e a sua dimensão e devolva a soma de todas as ocorrências do maior número do vector.

Exemplo de execução:

```
se vector: 1 2 8 8 1 8 → devolve: 24
se vector: 2 2 2 2 2 2 → devolve: 12
se vector: 1 2 3 2 -1 0 → devolve: 3
```

- b) (4.0 valores) Usando a função da alínea anterior, faça um programa que:
- declare um vector de inteiros de tamanho TAM
 - peça ao utilizador valores inteiros para preencher o vector
 - chame a função anterior enviando como argumentos o vector e a sua dimensão
 - imprima o valor devolvido pela função.

3.

(6.5 valores) Escreva uma função que receba uma frase, uma palavra e as dimensões dos respectivos vectores onde estão armazenadas as strings, como argumentos. A função deverá substituir a última palavra da frase pela palavra passada como 2º argumento, devolvendo 1 se a frase for, de facto, modificada e 0 caso contrário. Assuma que as palavras na frase estão separadas por um ou mais espaços, podendo existir espaços no início e no fim da frase.

O protótipo da função será:

```
int func(char frase[], char palavra[], int tamf, int tamp);
```

Considerando a seguinte função main()...

```
#include <stdio.h>
#define TAMFRASE 80
#define TAMPAL 15
void main()
{
    char f[TAMFRASE], p[TAMPAL];
    printf("Indique frase: ");    gets(f);
    printf("Indique palavra: ");  gets(p);
    if(func(f,p,TAMFRASE,TAMPAL))
        printf("Frase modificada:\n\t%s\n",f);
    else
        printf("Frase não modificada!\n");
}
```

... o resultado de execução deverá ser o seguinte:

```
Indique frase: Hoje é um dia importante
Indique palavra: normal
Frase modificada:
    Hoje é um dia normal
```

Nota: Caso necessite pode utilizar a função:

```
int strlen(char str[]);
```

a qual devolve o número de caracteres efectivos existentes na string "str" (sem contar com o '\0').