

# Exame de Programação Imperativa

(Duração: 2 horas)

Universidade do Algarve

15 de Janeiro de 2004

- Preenche o teu nome, nº de aluno e curso no topo de todas as folhas.
- Faz letra legível. Desliga o telemóvel.
- As perguntas 1 e 2 são de escolha múltipla com 4 opções: a) b) c) d). Faz uma cruz em cima da opção certa.
- O teste tem 9 perguntas e a cotação de cada uma aparece entre parêntesis.
- Boa sorte!

**(1 valor) Pergunta 1.** Qual o significado da declaração, `int *p;`

- a) p vai ser multiplicado por um inteiro
- b) p vai ser convertido para um inteiro
- c) p é uma variável que pode guardar um endereço de uma variável inteira
- d) p é um tipo de dados que pode ter endereços de variáveis inteiras

**(1 valor) Pergunta 2.** O que é que o seguinte pedaço de código escreve no ecrã?

```
int i;
float x;

i = 5;
x = 9 / (float) i;
i = (int) x;
printf("i=%d, x=%.2f", i, x);
```

- a) i=1.0, x=1.8
- b) i=1, x=1.8
- c) i=1, x=1.80
- d) i=1.0, x=1.80

**(2 valores) Pergunta 3.** A fórmula de Stirling permite fazer um cálculo aproximado do factorial de um número. A fórmula é dada pela seguinte expressão matemática:

$$n! \approx \sqrt{2 \pi n} \left(\frac{n}{e}\right)^n$$

Faça um programa que calcule o valor aproximado do factorial de um número através desta fórmula. Pode e deve utilizar as funções `sqrt`, `pow` e `exp`.

(Nota: `sqrt(x)` calcula  $\sqrt{x}$ , `pow(x,y)` calcula  $x^y$ , `exp(x)` calcula  $e^x$ ).

**(2 valores) Pergunta 4.** Imagine que a biblioteca `<string.h>` não tinha a função `'strlen'` definida. (Para os mais esquecidos, a função `strlen` devolve o número de caracteres de uma string). Implemente a função `strlen`.

**(2 valores) Pergunta 5.** Imagine que a linguagem C não tinha os operadores `+` e `-`. Desse modo, só poderíamos fazer somas e subtrações utilizando os operadores `++` e `--`. Por exemplo, para se calcular `5+3` teríamos de fazer qualquer coisa deste estilo:

```
int x;
x=5; x++; x++; x++;
```

Faça uma função chamada `'soma'`, que implemente esta funcionalidade. A função deve receber 2 inteiros e retornar um inteiro. (Não se esqueça de que os argumentos da função podem ser números negativos.)

**(1+1=2 valores) Pergunta 6.** Considera a seguinte função:

```
/* ... */
void misterio( int a, int b ) {
    if( a <= b ) {
        printf("%d ", a);
        misterio(a+1, b);
    }
}
```

- Substitui o nome `'misterio'` por um nome mais apropriado e substitui também as reticências por um comentário adequado.
- Implemente a mesma função sem ser de modo recursivo.

**(2 valores) Pergunta 7.** Escreva uma função em C que que receba um array de inteiros e a sua dimensão, e que apresente um histograma dos valores contidos no array. Para quem não sabe, um histograma é a mesma coisa que um gráfico de barras (ver exemplo). Por exemplo se a função for invocada com `a[]={2,4,7,10,6,2}` e `n=6`, o resultado deveria ser o seguinte:

```
0 |##
1 |####
2 |#####
3 |#####
4 |#####
5 |##
```

**(1+2+3=6 valores) Pergunta 8.** Considere a seguinte declaração em C:

```
typedef struct {
    char equipa1[30];
    char equipa2[30];
    int golos_equipa1;
    int golos_equipa2;
} resultado;

resultado jogo[9] = ...
```

- a) Inicialize a variável `jogo` com os resultados das 9 partidas realizadas na 16ª jornada da SuperLiga:

```
Belenenses 0-2 P.Ferreira
V.Guimarães 1-0 E.Amadora
Nacional 3-0 Alverca
Moreirense 3-1 Beira-Mar
Gil Vicente 2-1 Marítimo
Benfica 1-3 Sporting
Boavista 1-0 U.Leiria
Académica 0-1 Sp. Braga
FC Porto 1-0 Rio Ave
```

- b) Faça uma função que receba os resultados de uma jornada de futebol, e que retorne o número médio de golos por jogo.
- c) Faça uma função que receba os resultados de uma jornada de futebol, e que indique o(s) jogo(s) em que houve mais golos. Como exemplo, se a função recebesse como entrada a variável `'jogo'`, iria produzir com output:

```
Moreirense 3-1 Beira-Mar
Benfica 1-3 Sporting
```

**(1 valor) Pergunta 9.** Explique a vantagem de usar funções nos programas.