

# Ficha de trabalho – nº4

## Linguagem de Programação “C”

(Funções e passagem de parâmetros)

1. O programa pretende escrever no ecrã a frase “Olá pessoal”. Complete-o.

```
void diz_ola()
{
    ① _____;
}

main()
{
    diz_ola();
    getche ();
    return 0;
}
```

2. Considere a função:

```
int quad(int x)
{
    return x*x;
}
```

- 2.1. Indique o que faz.
- 2.2. Elabore um programa que pede ao utilizador para inserir um número inteiro e depois escreve no ecrã o quadrado desse número. Utilize a função **quad**.
3. Considere uma função que aceita como parâmetro dois números inteiros e devolve o maior dos números.
- 3.1. Implemente-a.
- 3.2. Implemente um programa que indica o maior de dois números inteiros inseridos pelo utilizador. O programa deve utilizar a função da alínea anterior.
- 3.3. Implemente um programa que indica o menor de dois números inteiros inseridos pelo utilizador. O programa deve utilizar a função **int menor(int a, int b)** .
4. Elabore um programa que resolve uma equação do 1º grau do tipo **Ax + B = C** em que o utilizador indica os valores dos coeficientes A, B e C. O programa deve utilizar a função **float Equa1Grau(float a, float b, float c)**.
5. Elabore um programa que resolve uma equação do 2º grau do tipo **Ax<sup>2</sup> + Bx + C = 0** em que o utilizador indica os valores dos coeficientes A, B e C. O programa deve utilizar a função **float Equa2Grau(float a, float b, float c)**.
6. Indique os erros de compilação que seriam detectados nas seguintes funções:

6.1.

```
int quad(int x);
{
    return x*x;
}
```

6.2.

```
void quad(int x)
{
    return x*x;
}
```

6.3.

```
int (int x)
{
    return x*x;
}
```

6.4.

```
int soma (int x, y)
{
    return x+y;
}
```

7. Considere o byte e os seus múltiplos. Resolva os seguintes exercícios:

- 7.1. 17 bytes = ? (bits)
- 7.2. 4,5 Mb = ? (bytes)
- 7.3. 59 Mb = ? (Kb)
- 7.4. 9 Gb = ? (Mb)
- 7.5. 13,9 Tb = ? (Gb)

8. Considere os sistemas de numeração decimal e binário. Resolva os seguintes exercícios de conversão entre os sistemas de numeração:

8.1.  $212_{(10)} = ?_{(2)}$

8.2.  $1\ 0111_{(2)} = ?_{(10)}$

8.3.  $257_{(10)} = ?_{(2)}$

8.4.  $1100\ 0001_{(2)} = ?_{(10)}$

8.5.  $2047_{(10)} = ?_{(2)}$

8.6.  $11\ 0110\ 0001_{(2)} = ?_{(10)}$

9. Elabore um programa que gere um número aleatório inteiro entre 1 e 100. O programa deve pedir ao utilizador para introduzir 2 números inteiros. O programa deve indicar se o número gerado está entre os dois números inteiros introduzidos pelo utilizador.

10. Considere as funções **ping** e **pong**.

Indique o output das seguintes chamadas:

10.1. `pong(3)`.

10.2. `ping(-4)`.

10.3. `ping(25)`.

10.4. `ping(2)`.

10.5. `ping(1)`.

```
int ping(int i)
{
    switch (i)
    {
        case 1:
        case 2:
        case 3:
            while (i>=0)
                printf("\n%d",--i);
            break;
        case 25:
            pong(3);
            break;
        default:
            printf("\nJa passei a LP");
            pong(123);
    }
}

void pong(int x)
{
    int j=0;

    switch (x)
    {
        case 1:
        case 2: ping(x);
        case 3: j=5;
                j++;
                return;
        default:
            printf("\nOla");
            return;
    }
    printf("\nVou sair");
}
```

11. Elabore uma função (GeraNumero) que gera aleatoriamente um número inteiro pertencente a um intervalo de valores indicado no parâmetro da função. Por exemplo:

```
int n;
n = GeraNumero(1,9); //neste caso, n poderá ter um número inteiro entre 1 e 9 incluindo os limites.
```

12. Considere um programa simule o sorteio do euromilhões. O programa deve gerar 5 números entre 1 e 50 e 2 números (as estrelas) entre 1 e 9. Elabore-o usando a função criada no exercício anterior.

13. O programa seguinte pretende escrever no ecrã as 5 notas que estão no vector **nota**. Complete o programa.

```
int main()
{
    int i, nota[5];

    nota[ ① _____ ]=13;nota[1]=11;nota[2]=8;nota[3]=16;nota[4]=19;

    for (i=0;i< ② _____ ;i++)
    {
        printf("%d\n",nota[ ③ _____ ]);
    }

    return 0;
}
```

14. Codifique um programa que declare um vector para armazenar 5 notas de um teste de LP. O programa deve mostrar:

- a média das notas;
- o número de positivas;
- o número de negativas;
- a percentagem de positivas e negativas;
- a melhor nota;
- a pior nota;
- o número de notas acima e abaixo da media;