



**Escola de Engenharia da Universidade do Minho**  
Mestrado Integrado em Eng. Electrónica Industrial e Computadores  
**Programação de Computadores**

2012/2013  
MIEEIC  
(1º Ano)  
1º Sem

**Exame 1, Duração 1h45m - Com Consulta**

Nome: \_\_\_\_\_ Nº Aluno.: \_\_\_\_\_

Responda às seguintes questões, preenchendo a tabela com a **opção correcta (em maiúsculas)** (Correcto: x Val / Errado: -x/3 Val).  
Suponha que foram realizados as inclusões das bibliotecas necessárias (exemplo: #include <stdio.h> e #include <stdlib.h>).

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Suponha a seguinte função em linguagem C:

```
void f1(int a, int b)
{
    if(a>b && a>5 || b>5) printf("1");
    else if(b>6 || a<3) printf("2");
    else { printf("3");
          printf("4"); }
}
```

1) Se executar a chamada f1(6,6) o que é escrito no ecrã?

a) 1    b) 14    c) 2    d) 34

e) Nenhuma das Anteriores

2) Se executar a chamada f1(2,5) o que é escrito no ecrã?

a) 2    b) 3    c) 24    d) 34

e) Nenhuma das Anteriores

Suponha a seguinte função em linguagem C:

```
void f2(int op) {
    switch(op) {
        case 1: printf("1");
                printf("2");
                break;
        case 2: break;
        case 3: printf("3");
        case 4: printf("4"); break;
        default: printf("5");
    }
}
```

3) Se executar a chamada f2(2) o que é escrito no ecrã?

a) 2    b) 34    c) 234    d) 2345

e) Nenhuma das Anteriores

4) Se executar a chamada f2(1) o que é escrito no ecrã?

a) 1234    b) 1    c) 12    d) 12345

e) Nenhuma das Anteriores

Suponha a seguinte função em linguagem C/C++:

```
void ciclos(int n1, int n2)
{
    for(int i=1; i<n1; i++)
        for(int j=n2; j<i+3; j*=2)
            printf("%d-%d ", i, j);
}
```

5) Executando ciclos(4,1) o que é escrito no ecrã?

a) 1-1 1-2 2-1 2-2 2-4

b) 1-1 1-2 1-3 2-1 2-2 2-3 3-1 3-2 3-3

c) 1-1 1-2 2-1 2-2 2-4 3-1 3-2 3-4

d) 1-1 1-2 1-3 1-4 1-5

e) Nenhuma das Anteriores

6) Executando ciclos(6,2) o que é escrito no ecrã?

a) 1-2 2-2 2-4 3-2 3-4 4-2 4-4 5-2 5-4

b) 1-2 2-2 3-2 4-2 5-2

c) 1-2 2-2 3-2 4-2 5-2 6-2

d) 1-2 2-2 2-3 3-2 3-3 4-2 4-3 5-2 5-3

e) Nenhuma das Anteriores

Suponha a seguinte função em linguagem C/C++:

```
void ciclos2(int N, int no)
{
    for(int i=0; i<N; i++) {
        for(int j=0; j<=no; j++)
            if(i>j+1) printf("O");
            else printf("*");
        printf("\n");
    }
}
```

7) Executando ciclos2(4,2) o que é escrito no ecrã?

a)	b)	c)	d)
***	****	****	****
***	****	****	O***
***	****	O***	OO**
***		OO**	OOO*

e) Nenhuma das Anteriores

8) Executando ciclos2(5,1) o que é escrito no ecrã?

a)	b)	c)	d)
OO	**	**	**
OO	**	**	O*
OO	**	O*	OO
OO	**	OO	OO
OO	**	OO	OO

e) Nenhuma das Anteriores

9) Indique o que faz a função xpto1?

```
void xpto1(int v[10])
{
    int aux;
    for(int i=0; i<=9; i++) {
        aux=v[i]; v[i]=v[9-i]; v[9-i]=aux; }
    return;
}
```

a) Calcula a soma dos elementos pares do vetor v[].

b) Não tem qualquer efeito.

c) Inverte a ordem dos elementos do vetor v[].

d) Ordena o vector v[] por ordem crescente.

e) Nenhuma das Anteriores.

10) Supondo o seguinte programa em linguagem C/C++ indique o que é escrito no ecrã.

```
int main(void){
    int *ap, x, y, z;
    x = 10; ap = &x; *ap = 20;
    y = 2*x; z = 2*(*ap);
    printf("x=%d y=%d z=%d \n",
        x,y,z);
    system("PAUSE"); return 0;
}
```

- a) O programa dá erro na compilação
- b) x=20 y=20 z=20
- c) x=20 y=40 z=40
- d) x=40 y=40 z=40
- e) Nenhuma das Anteriores

11) Substituindo a instrução `*ap=20;` por `ap=20;` no programa anterior indique o que é escrito no ecrã.

- a) O programa dá erro na compilação
- b) x=20 y=20 z=20
- c) x=20 y=40 z=40
- d) x=40 y=40 z=40
- e) Nenhuma das Anteriores

12) Supondo o seguinte programa em linguagem C/C++ indique o que é escrito no ecrã.

```
void xpto(int a, int b, int c){
    a = 1; b = 2; c = 3;
}
int main(void)
{
    int x = 5, y = 10, z = 20;
    xpto(&x, &y, &z);
    printf("%d %d %d", x, y, z);
}
```

- a) 1 10 20
- b) 5 2 3
- c) 1 2 3
- d) Nada pois o programa dá erro na compilação
- e) Nenhuma das Anteriores

13) Diga o valor das variáveis x, y, z e w depois da seguinte sequência de operações:

```
int x,y,z,w; x=10; y=10;
z=x++; x=--x+1; w=++y;
printf("%d %d %d %d\n",x,y,z,w);
```

- a) x=11 y=11 z=11 w=10
- b) x=11 y=11 z=10 w=11
- c) x=-11 y=11 z=10 w=11
- d) x=-10 y=10 z=10 w=11
- e) Nenhuma das Anteriores

14) Substituindo a instrução `x=--x+1;` por `x=x--;` no programa anterior indique o que é escrito no ecrã

- a) x=11 y=11 z=11 w=10
- b) x=11 y=11 z=10 w=11
- c) x=-11 y=11 z=10 w=11
- d) x=-10 y=10 z=10 w=11
- e) Nenhuma das Anteriores

15) Supondo o seguinte programa em linguagem C/C++ indique o que é escrito no ecrã.

```
int main(void) {
    for(int i=1; i<=3; i++)
        printf(" %0.*d ", i, 98);
    system("PAUSE"); return 0;
}
```

- a) 98 98.0 98.00
- b) 98 98 098
- c) 98 098 0098
- d) 98.0 98.00 98.000
- e) Nenhuma das Anteriores

16) Construa uma função `void calculos(int a, double x)` que receba dois números (um inteiro e um real) e escreva no écran (sempre utilizando campos com 10 casas (nos exemplos os '.' representam espaços): os dois números; a divisão (real) dos números por 5 (com duas casas decimais); a divisão inteira dos números por 5 e os números formatados, com a formatação apresentada abaixo (colocando 1 a 5 zeros à esquerda e colocando 1 a 5 casas decimais).

Exemplos: <code>calculos(2, 9.5)</code>	<code>calculos(10,15.2344)</code>
.....2.....9.50	.....10.....15.23
.....0.40.....1.90	.....2.00.....3.05
.....0.....1	.....2.....3
.....2.....9.5	.....10.....15.2
.....02.....9.50	.....10.....15.23
.....002.....9.500	.....010.....15.234
.....0002.....9.5000	.....0010.....15.2344
.....00002.....9.50000	.....00010.....15.23440

17) Construa a função `void pira(int alt)` que dada a altura `alt` desenhe pirâmides compostas de símbolos \* e O, com aspeto semelhante ao apresentado abaixo, de acordo com os exemplos apresentados:

<code>pira(1)</code>	<code>pira(2)</code>	<code>pira(3)</code>	<code>pira(5)</code>
**	**	**	**
	*OO*	*OO*	*OO*
		*OOO*	*OOO*
			*OOOO*
			*OOOOOO*
			*OOOOOOOO*

Suponha que dispõe da seguinte função já implementada:

```
void nc(int n, char c) {
    for(int i=0; i<n; i++) printf("%c", c); }
```

18) Escreva uma função `bool pede_data(int *dia, int *mes, int *ano)` que peça uma data ao utilizador (dia, mês e ano) e verifique se a data é válida. Na resolução do exercício siga as seguintes regras:

- Considere que o ano é válido se for igual ou inferior a 2012.
- Ignore os anos bissextos considerando que Fevereiro tem sempre 28 dias.
- Utilize a instrução `switch ... case`, na função, para determinar o número de dias do mês.
- Caso a data seja válida retorne verdadeiro e os valores correspondentes do dia, mês e ano.
- Caso a data seja inválida retorne falso e 0, 0, 0 para o dia, mês e ano.

19) Construa uma função `int inverte(int v1[], int n1, int val1, int val2)` que dado um vetor de inteiros inverta o vetor e, ao mesmo tempo, substitua todos os elementos iguais a `val1` por `val2` retornando o número de elementos que foram substituídos. Por exemplo, supondo o seguinte vetor inicial:

`v1[11]={1,2,3,5,7,3,8,9,9,3,4}`, a chamada de `inverte(v1,11,3,10)` retorna o valor 3 e o vetor ficará com os valores: `v1[11]={4,10,9,9,8,10,7,5,10,2,1}`

20) Suponha uma matriz, já inicializada `tabul[N][N]` contendo números inteiros representando peças e damas de dois jogadores (1 - peça jogador 1, 2 - peça jogador 2, 3 - dama do jogador 1, 4 - dama do jogador 2) e espaços vazios (0) num tabuleiro com dimensão  $N \times N$ . Construa uma função `int move_pecas1(int tabul[N][N])` que dada a matriz com o estado do jogo movimente todas as peças e damas do jogador 1 uma casa na horizontal para baixo. Suponha que as peças (e damas) do jogador 1 movem-se na vertical para baixo, uma casa de cada vez. As peças só se podem movimentar para casas vazias ou que sejam esvaziadas no decurso da movimentação global (caso contrário permanecem imóveis). Caso as peças atinjam o fim do tabuleiro (parte inferior do tabuleiro), e ainda não sejam damas, transformam-se em damas. A função deve retornar o número de peças efetivamente movimentadas. *Sugestão: Comece pela linha de baixo (N-1) verificando que pelas se podem movimentar essa linha e vá subindo, acabando na segunda linha (linha 1).*

Exemplo: Aplicando a função `move_pecas1(tabul)`, à matriz da esquerda é retornado o valor 5 (cinco peças movimentadas) e muda a matriz para a matriz representada à direita:

Matriz Inicial:	Resultado Final:
1 1 0 0 1	0 1 0 0 0
1 4 0 1 0	1 4 0 1 1
0 0 0 2 0	1 0 0 2 0
1 0 2 0 3	0 0 2 0 0
0 0 3 2 0	3 0 3 2 3

21) Construa um programa principal que lhe permita testar, com exemplos relevantes, os programas construídos nas alíneas 16, 17, 18, 19 e 20.