

*Notas:*

1. *Cotação de cada pergunta:* 1. 4.5 / 2. 4.5 / 3. 5.5 / 4. 5.5
2. Responda às questões de forma clara e concisa nas folhas de exame distribuídas.

1. Em cada uma das alíneas, diga o que escreve cada fragmento de código dado.

- a) 

```
int x= 5, y=10, z=20;
if (x*2==y && y/z==0)
    printf("%d %d\n",y+2*x,z-y);
else
    printf("%d\n",x+y);
```
- b) 

```
int main() {
    int i, n= 3;
    int v[3][3] = {{1,2,3},
                   {1,2,3},
                   {1,2,3}};
    for (i= 0; i < n; i++)
        for (j=i; j<n ; j++)
            printf("%d ",v[i][j]);
    return 0;
}
```
- c) 

```
int main() {
    char a='s', b='n';
    printf("%c %c\n", h(&a,b), a, b);
    return 0;
}

char h(char *a, char b) {
    if(*a=='s') {
        *a='t';
        b='x';
        return 'f';
    }
    else return 'v';
}
```

## 2. Strings

Pretende-se processar strings que contêm resultados de jogos de futebol com o seguinte formato "Unidos-2, Solteiros-1". O número de golos tem sempre exatamente um dígito. Responda às seguintes questões:

- a) Escreva uma função `void leResultado(char eq1[], char eq2[], char golos[])` que leia para o vector `eq1` o nome da primeira equipa, para `eq2` o nome da segunda e para `golos` o resultado (no caso do exemplo a string "21". Uma linha de input é terminada pelo carácter mudança de linha ou pelo fim de input. Não poderá fazer uso de funções de sistema de manipulação de strings.
- b) Escreva uma função `char comenta(char eq1[], char eq2[], char res[])` que analise a linha lida e que comente o resultado de forma idêntica aos exemplos abaixo:

Unidos venceram Solteiros por 2 a 1.    Em caso de vitória.  
Unidos e Solteiros empataram a 3.        Em caso de empate.

Assume-se sempre que os nomes das equipas é um plural. Qualquer uma das equipas pode vencer o jogo. Pode haver um resultado "Aliados-0, Casados-4". A função

retorna 'x' em caso de empate, '1' em caso de vitória da primeira equipa e '2' em caso de vitória da segunda.

Exemplos:

Entrada	Saída	Retorna
Ases-1, Gaiatos-1	Ases e Gaiatos empataram a 1.	x
Artes-2, Cotas-0	Artes venceram Cotas por 2 a 0.	1
Muros-1, Kraks-3	Kraks venceram Muros por 3 a 1.	2

### 3. Vectores/Matrizes

Pretende-se agora guardar os resultados de um campeonato de futebol. As equipas participantes são guardadas no vetor de strings `char *equipas[]`. Os resultados dos jogos entre as equipas são guardados numa matriz quadrada de inteiros `int res[][]`. Cada linha da matriz representa os resultados em casa, em termos de 1, 0, 2 de uma equipa contra todas as outras. O 1 representa a vitória em casa, 2 a vitória fora e 0 o empate. Na diagonal principal a matriz tem zeros. Cada duas equipas fazem dois jogos uma conta a outra. Um em casa (no campo) da primeira, outro em casa da segunda.

```
equipas[0] = "Ases"    res[0][] = {0,1,1,2,0}
equipas[1] = "Broas"   res[1][] = {1,0,0,2,2}
...
```

- a) Escreva uma função `void leResults(int ne, char equipas[][MAXCH], int res[][MAXEQ])` que lê do ecrã a informação sobre as `ne` equipas, i.e. o nome e os resultados que cada equipa obteve em casa contra as outras. Não são dados os valores para a diagonal principal pois já se sabe que são zeros. Para 5 equipas, os resultados são dados assim:

```
Ases 1 1 2 0
Broas 1 0 2 2
...
```

- b) Escreva uma função `int pontosGanhos(int ne, int ieq, int res[][MAXEQ], int Casa)` que, se o valor do argumento `Casa` for 1, calcula os pontos que a equipa ganhou em casa (resultados da linha). A equipa é indicada pelo seu índice `ieq` no vetor `equipas`. Uma vitória vale 3 pontos, um empate vale 1 ponto e uma derrota vale 0 pontos. No caso de `Casa==0` a função retorna os pontos ganhos fora. No exemplo acima, os Ases ganham 7 pontos em casa (duas vitórias, um empate e uma derrota). Os Broas ganham 4.

- c) Escreva uma função que, dados a lista das equipas e os resultados, apresente no output uma descrição, para cada equipa, dos resultados em termos de pontos ganhos e de derrotas caseiras (neste caso listando os nomes dos adversários que a conseguiram vencer no seu próprio terreno). No caso de não haver derrotas caseiras a registar, escreve-se *Nenhuma*. Deve usar as funções pedidas nas alíneas anteriores mesmo que não as tenha definido.

#### Análise detalhada

Ases: pontos em casa - 7, pontos fora - 3, total - 10.

Derrotas caseiras:

Aliados

Broas: pontos em casa - 4, pontos fora - 1, total - 5.

Derrotas caseiras:

Aliados

Unidos

Aliados: pontos em casa 10, pontos fora - 6, total - 16.

Derrotas caseiras:

Nenhuma

...

- d) Descreva o método da seleção aplicado à ordenação descendente do vetor {3, 1, 4, 5, 2}. Implemente uma função que, dados dois vetores com os pontos que cada equipa ganhou em casa (`int casa[]`) e os pontos que ganhou fora (`int fora[]`), o vetor onde serão registadas as posições na tabela classificativa de cada equipa (`int pos[]` que inicialmente tem os valores 0, 1, 2, 3, etc.) assim como o número de equipas, ordene as posições no vetor `pos` de forma a que a primeira equipa seja a que tem o maior total de pontos e assim sucessivamente. Em caso de empate usam-se os pontos obtidos fora como critério de desempate. Assuma que após a aplicação destes dois critérios não haverá empates.

#### 4. Estruturas

Chegou-se à conclusão que uma boa forma de representar o resultado de um jogo era usando estruturas. Em baixo vemos a estrutura utilizada.

```
typedef struct {
```

```

char eq1[]; // Nome da equipa que joga em casa
int golos1; // Golos que marcou a equipa da casa
char eq2[]; // Nome do visitante
int golos2; // Golos que marcou o visitante
} RESULTADO;

```

A título de exemplo, uma variável `r` que represente o jogo entre os Unidos e os Gunas, que ficou 2 a 5, declara-se assim:

```
RESULTADO r= {"Unidos",2,"Gunas",5};
```

Responda às seguintes questões:

- Escreva uma função `int eliminatoria(RESULTADO r1, RESULTADO r2)` que diz qual o resultado de uma eliminatória a duas mãos. Numa eliminatória a duas mãos entre duas equipas joga-se primeiro em casa de uma e depois em casa da outra. Passa a equipa que, na soma dos dois jogos, marcar mais golos. Em caso de empate, passa a equipa que marcou mais golos fora. Se o empate persistir, recorre-se à marcação de penaltis. A função retorna 1 se vencer a primeira equipa a jogar em casa, 2 se vencer a segunda e 0 se tiverem de ir a penaltis. Por exemplo, com os resultados “Ases-5, Tios-2” e “Tios-4, Ases-1” os Tios passam e a função retorna 2. Com “Ases-1, Tios-0” e “Tios-2, Ases-1” passam os Ases e retorna 1. Se os resultados forem iguais mas em sentidos opostos retorna 0.
- Escreva uma função `int DifGolos(RESULTADO res[],int nj)` em que dado um vetor `res` de `nj` resultados realizados em casa pela mesma equipa retorne a diferença total entre golos marcados e golos sofridos dessa equipa.
- Escreva uma função `RESULTADO paraPassar(RESULTADO r)` que, dado um resultado de uma primeira mão de uma eliminatória entre duas equipas retorna o resultado mínimo, sem perder, que a equipa que joga fora no primeiro jogo tem que fazer na segunda mão para passar a eliminatória. Por exemplo, se `r` tem o resultado “Broas-2, Aliados-1”, aos Aliados basta vencer o segundo jogo por 1 a 0. Se for “Broas-2, Aliados-0” os Aliados são obrigados vencer em casa por 3 a 0. Se o primeiro resultado for “Broas-1, Aliados-1”, outro empate com golos ou uma vitória dos Aliados basta o 0 a 0. Se o primeiro resultado for 0 a 0, então deve vencer por 1 a 0.