

EXAME NACIONAL DO ENSINO SECUNDÁRIO
12.º Ano de Escolaridade (Decreto-Lei n.º 286/89, de 29 de Agosto)
Curso Tecnológico de Informática

Duração da prova: 120 minutos
2005

1.ª FASE

**PROVA ESCRITA DE ESTRUTURA, ORGANIZAÇÃO
E TRATAMENTO DE DADOS**

A prova é constituída por quatro grupos.

Os três itens do Grupo I são relativos a algoritmia.

Dois dos três itens do Grupo II são relativos a redes PERT.

Os três itens do Grupo III são relativos a gestão de bases de dados.

Os itens do Grupo IV são relativos a programação em Visual Basic.

Todos os itens exigem respostas de composição curta, à excepção do terceiro item do Grupo II, que é um item de desenvolvimento.

GRUPO I

Considere uma aplicação de um editor de texto básico.

1. Sugira uma estrutura de dados (vector, matriz ou ficheiro) que permita guardar, em memória principal, as palavras de um texto com um máximo de 1000 linhas. Considere que cada linha tem, no máximo, 100 palavras. Apresente, no caso de a estrutura de dados escolhida ser um vector ou uma matriz, a sua dimensão; no caso de ser um ficheiro, o nome dos campos constituintes dos seus registos.
2. Transcreva para a sua folha de prova os termos adequados ao preenchimento dos espaços (①, ②, ... , ⑤) assinalados no algoritmo seguinte.

Considere a existência de dois vectores: um, designado por **FRASE**, contém os **N** caracteres que constituem uma frase, e um outro, designado por **PROCURA**, contém 10 caracteres. Pretende-se fazer uma contagem do número de vezes que cada carácter do vector **PROCURA** ocorre no vector **FRASE**, guardando o resultado no vector **CONTAGEM**. Assim, no vector **CONTAGEM**, na posição **I**, ficará o número de vezes que o carácter **PROCURA[I]** aparece no vector **FRASE**.

Consideram-se os vectores **FRASE** e **PROCURA** previamente preenchidos.

1.[Inicializações]

Repita para $I = 1, 2, \dots, 10$

① _____ $\leftarrow 0$

2.[Preenchimento do vector CONTAGEM]

Repita para $I = 1, 2, \dots, N$

Repita para $J = 1, 2, \dots, ②$ _____

Se ③ _____ = **PROCURA[J]**

Então **CONTAGEM[J]** \leftarrow ④ _____

3.[Imprimir o resultado]

Repita para $I = 1, 2, \dots, 10$

Escrever('O caracter', ⑤ _____, 'aparece', **CONTAGEM[I]**, 'vezes')

4.[Terminar]

Saída

3. Elabore o **passo 2** do algoritmo seguinte, para que, a partir do vector **LISTA** com **M** caracteres, imprima para o ecrã o número de ocorrências de pares de caracteres iguais e em posições consecutivas. Por exemplo, para $M=10$ e para a sequência de caracteres **LISTA**, {'a','a','a','b','b','b','b','f','h','h'}, o valor impresso no ecrã seria 4. Considere que um determinado carácter só pode fazer parte de um par.

1.[Ler o vector LISTA]

Repita para $I = 1, 2, \dots, M$

Leia(LISTA[I])

2.[Contar o número de ocorrências de pares de caracteres iguais e em posições consecutivas]

...

GRUPO II

1. Elabore a rede PERT correspondente à tabela seguinte.

Actividade	Actividades precedentes
A	—
B	—
C	A, B
D	B
E	B
F	D, E
G	D, E
H	G
I	D, H
J	D
K	C, I, J, F

2. Tendo como base a rede PERT seguinte (Fig. 1), representativa das actividades a desenvolver num projecto e das respectivas durações em dias, determine o(s) caminho(s) crítico(s) e o tempo crítico da realização do projecto.

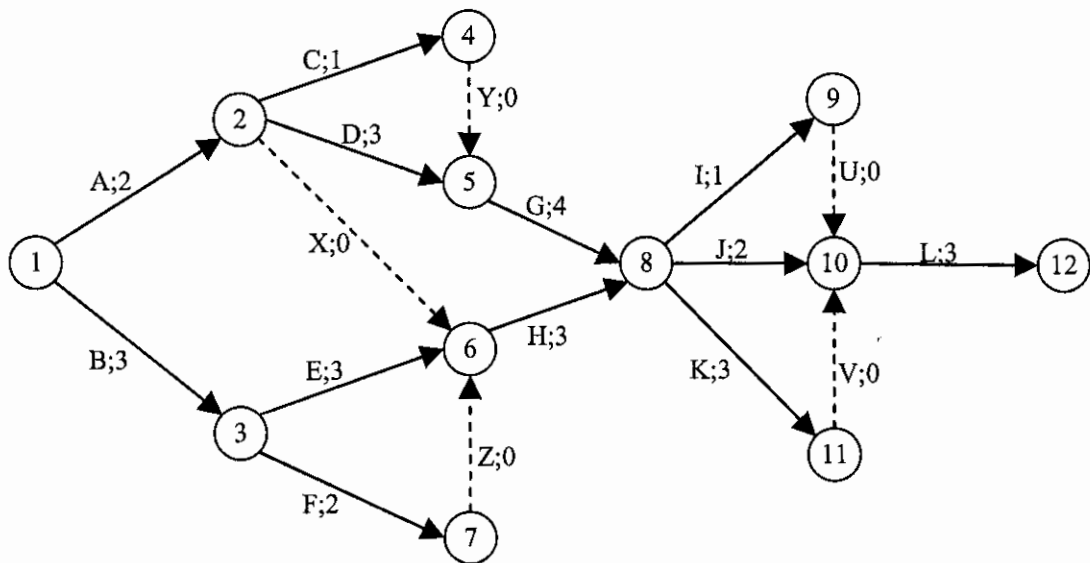


Fig. 1 – Rede PERT

3. Refira sucintamente dois efeitos que posturas incorrectas do corpo podem provocar, a médio/longo prazo, na saúde de um utilizador intensivo de meios informáticos, indicando quais as posturas que estão na origem do problema.

GRUPO III

1. Identifique e transcreva para a sua folha de prova os termos adequados ao preenchimento dos espaços assinalados abaixo (①,②,...,⑤), num texto relativo aos Sistemas Gestores de Bases de Dados (SGBD).

As consultas criadas no Access através da ferramenta gráfica QBE (Query by Example) correspondem a instruções equivalentes, criadas numa linguagem cujas siglas são SQL. Uma consulta de selecção corresponde a uma instrução que se inicia com a cláusula ① _____. Já uma consulta de inserção inicia-se com a cláusula ② _____, e uma consulta de actualização, com a cláusula ③ _____. Nas consultas de selecção, se existirem critérios de selecção, virão imediatamente a seguir à cláusula ④ _____ e, quando são definidos vários critérios que não necessitem de se verificar simultaneamente, usa-se o operador lógico ⑤ _____.

2. Defina um conjunto de tabelas (nomes e respectivos campos) a incluir numa base de dados que dê suporte ao formulário seguinte (Fig. 2), respeitando o modelo relacional (tabelas normalizadas).

Num_Atendimento	Data	Hora	Duração	Resumo	Cod_Operador	Operador
1	01/Jan/05	2:00	10	Assistência	1	Maria Dias
2	04/Fev/05	9:23	20	Configuração	2	José Alves
3	05/Fev/05	12:30	15	Informações	1	Maria Dias

Fig. 2 – Formulário de atendimentos a clientes

Nota: Considere que a numeração atribuída a cada atendimento (Num_Atendimento) é feita de forma sequencial para cada cliente.

3. Considere as seguintes tabelas (Fig. 3), relativas a uma base de dados referente a uma rede de armazéns e aos produtos existentes em cada armazém.

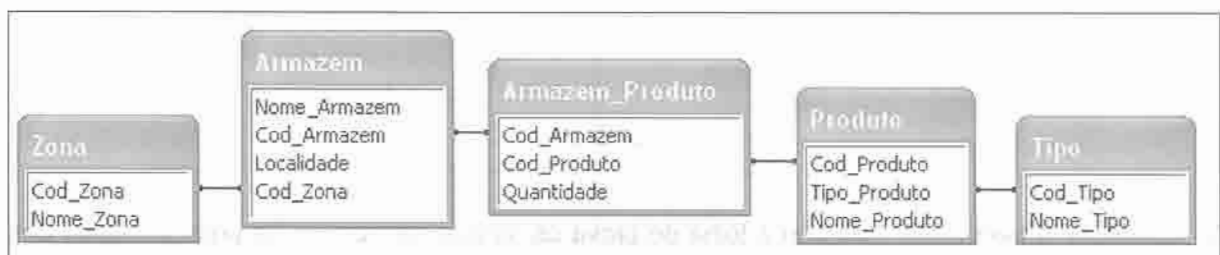


Fig. 3 – Tabelas da base de dados

- Notas:**
- Os campos «Cod_Zona», «Cod_Armazem», «Cod_Produto», «Cod_Tipo», «Tipo_Produto» e «Quantidade» são numéricos.
 - Os campos restantes são alfanuméricos.
 - Os valores dos campos «Cod_Armazem» e «Cod_Produto» são atribuídos sequencialmente.

3.1. Apresente, para cada uma das tabelas (Fig. 3), o campo, ou a combinação de campos, que deve constituir a sua chave primária.

3.2. Identifique o objectivo das consultas seguintes (Fig. 4 e Fig. 5), construídas mediante a utilização de uma ferramenta gráfica convencional de elaboração de consultas, discriminando, caso existam, tabelas usadas, campos de saída, campos de actualização, critérios de selecção, expressões, agrupamento de dados e junção de tabelas.

3.2.1.

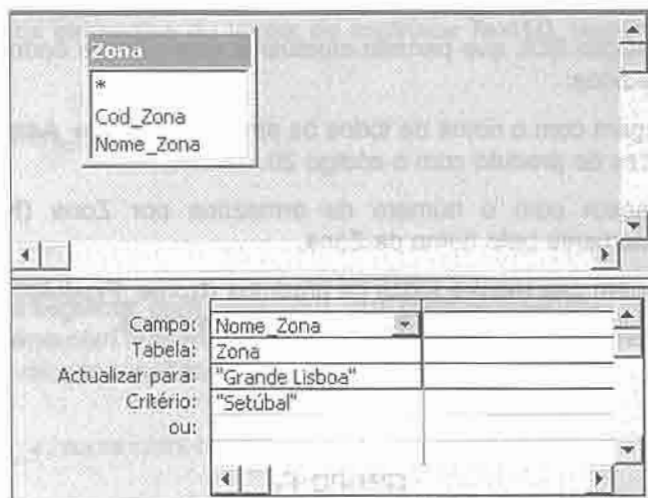


Fig. 4 – Consulta construída com recurso a ferramenta gráfica

3.2.2.

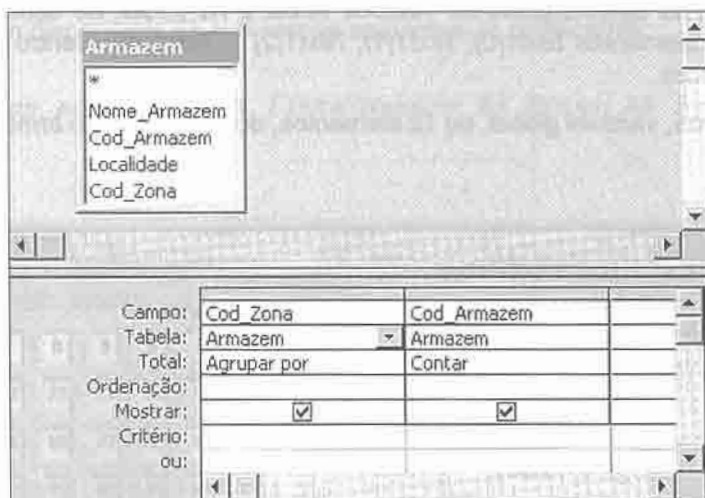


Fig. 5 – Consulta construída com recurso a ferramenta gráfica

3.3. Descreva o objectivo de cada uma das instruções SQL seguintes, discriminando, caso existam, tabelas usadas, campos de saída, campos de actualização, critérios de selecção, expressões, agrupamento de dados e junção de tabelas.

3.3.1. `SELECT Tipo_Produto, Nome_Produto
FROM Produto
GROUP BY Tipo_Produto, Nome_Produto`

3.3.2. `SELECT Nome_Produto, Min(Quantidade)
FROM Produto, Armazem_Produto
WHERE Produto.Cod_Produto = Armazem_Produto.Cod_Produto
GROUP BY Nome_Produto`

```

3.3.3. SELECT Nome_Armazem, Sum(Quantidade),Tipo_Produto
FROM Produto, Armazem, Armazem_Produto
WHERE Armazem.Cod_Armazem=Armazem_Produto.Cod_Armazem AND
Produto.Cod_Produto=Armazem_Produto.Cod_Produto
GROUP BY Nome_Armazem,Tipo_Produto

```

3.4. Apresente a instrução SQL que permite efectuar a selecção de dados adequada a cada um dos seguintes pedidos:

3.4.1. Uma listagem com o nome de todos os armazéns (Nome_Armazem) que têm mais de 20 unidades do produto com o código 20.

3.4.2. Uma listagem com o número de armazéns por Zona (Nome_Zona), ordenada ascendentemente pelo nome da Zona.

3.4.3. Uma listagem que mostre todos os produtos (Nome_Produto) e todas as unidades de cada produto existentes em toda a rede de armazéns.

GRUPO IV

Considere o seguinte *Form* de Visual Basic (Fig. 6), que permite gerar um conjunto de 6, 7, 8, 9, 10, 11 ou 12 números aleatórios apresentados na *TextBox Text2* e no vector de controlos do tipo *TextBox Text1()* que contém os elementos *Text1(0)*, *Text1(1)*, *Text1(2)* ... *Text1(48)*, dentro da *Frame* identificada por «Números Sorteados».

No vector **ListaNumeros**, variável global, de 12 elementos, do tipo *byte*, são armazenados os números aleatórios gerados.



Fig. 6 – Formulário em Visual Basic

1. No formulário apresentado, o botão que está activo é o identificado por «Gerar Números». Que método, associado ao botão identificado por «Limpar», deverá ser invocado para que, na activação do formulário, o botão fique activo?

2. Descreva a funcionalidade representada pelo sublinhado da letra «L» (Limpar) da propriedade **caption** do botão **CmdBtn_Limpar** e mostre como a implementaria.
3. Codifique a sub-rotina seguinte, relativa ao evento Click do botão **CmdBtn_Limpar**, que atribui o valor **vbWindowBackground** à propriedade **BackColor** e o valor **vbWindowText** à propriedade **ForeColor** de todos os elementos do vector de controlos **Text1()**, limpando ainda o conteúdo da **TextBox Text2**.

```
Private Sub CmdBtn_Limpar_Click()  
    ...  
End Sub
```

4. Codifique a sub-rotina seguinte, **gera_aleatorios**, que permite preencher os primeiros **N** elementos do vector **ListaNumeros** com números aleatórios, ordenados numericamente por ordem crescente, não devendo utilizar valores repetidos.

```
Public Sub gera_aleatorios(N As Byte)  
    ...  
End Sub
```

Considere que tem disponíveis as sub-rotinas seguintes, já codificadas:

```
Public Function encontrou_na_lista(Numero As Byte) As Byte  
    ...  
End Function
```

que verifica se o valor da variável **Numero** existe no vector **ListaNumeros()**, retornando a 1, caso exista, e a 0 caso não exista.

e

```
Public Sub Ordena(NumElementos As Byte)  
    ...  
End Sub
```

que ordena os primeiros **NumElementos** elementos do vector **ListaNumeros()**.

FIM

COTAÇÕES

Grupos	Questões	Cotações	Total das cotações por grupo
I	1.	5	25
	2.	10	
	3.	10	
II	1.	15	45
	2.	15	
	3.	15	
III	1.	15	90
	2.	20	
	3.1.	10	
	3.2.1.	6	
	3.2.2.	9	
	3.3.1.	5	
	3.3.2.	5	
	3.3.3.	5	
	3.4.1.	5	
	3.4.2.	5	
	3.4.3.	5	
IV	1.	5	40
	2.	5	
	3.	10	
	4.	20	
TOTAL			200