

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
2001

2.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.
- O único item do Grupo IV é relativo 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.

V.S.F.F.

113/1

GRUPO I

Considere uma aplicação relativa ao registo de acidentes de viação.

1. Sugira uma estrutura de dados (vector, matriz ou ficheiro) que permita guardar, em memória primária, o número de acidentes registados em cada um dos **12** meses do ano. No caso de a estrutura de dados escolhida ser um vector ou matriz, apresente a sua dimensão; no caso de ser um ficheiro, apresente 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.

Este algoritmo, com base nos valores da matriz **ACIDENTES** (previamente preenchida com o número de acidentes registados em **17** distritos, nos últimos **3** anos), lista, por ordem descendente, os **5** distritos com maior número de acidentes no somatório dos **3** anos considerados. Assume-se que não existem distritos com o mesmo número de acidentes no somatório dos **3** anos.

1. [Determinar os 5 distritos com o maior número de acidentes]

Repita para $I=1,2,\dots,①$ _____

TOTAL ← 0

Repita para $J=1,2,3$

TOTAL ← TOTAL + ACIDENTES[I, J]

$K \leftarrow 1$

Repita enquanto TOTAL < TOP[K, 2] e $K < 5$

$K \leftarrow K + 1$

Se TOTAL > TOP[② _____, 2]

Então Repita para $Z=5,4,\dots,③$ _____

TOP[Z, 1] ← TOP[Z-1, 1]

TOP[Z, 2] ← TOP[Z-1, 2]

TOP[K, 1] ← I

TOP[K, 2] ← ④ _____

2. [Listar, por ordem descendente, os 5 distritos com o maior número de acidentes]

Repita para $K=1,2,\dots,5$

Escreva ('No distrito ', TOP[K, 1], ' foram registados ', TOP[K, ⑤ _____])

3. [Terminar]

Saída

3. Elabore o **passo 3** do algoritmo seguinte, tendo em conta as seguintes considerações:
 - existem 3 categorias de veículos, «Ligeiros», «Pesados» e «Motociclos», identificadas, respectivamente, pelos números 1, 2 e 3;
 - cada acidente envolve 2 veículos com uma determinada categoria;
 - cada elemento da matriz **ACIDENTES_CATEGORIAS**[I, J] acumula o número de acidentes registados entre veículos da categoria **I**, com veículos da categoria **J**;
 - as variáveis numéricas **V1_CATEGORIA** e **V2_CATEGORIA** permitem introduzir as categorias dos 2 veículos;
 - a introdução de dados termina quando o utilizador digita **9** na variável **V1_CATEGORIA**.

1. [Ciclo de leitura de dados e preenchimento da matriz]

CATEGORIA ← 0

Repita o passo 2 e 3 enquanto $V1_CATEGORIA \neq 9$

2. [Ler os dados relativos ao acidente]

Leia(V1_CATEGORIA)

Leia(V2_CATEGORIA)

3. [Preencher a matriz]

...

GRUPO II

1. Considere a tabela que se segue. Elabore a rede PERT correspondente.

Actividade	Actividades precedentes
A	-
B	-
C	-
D	A
E	C
F	D
G	E

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.

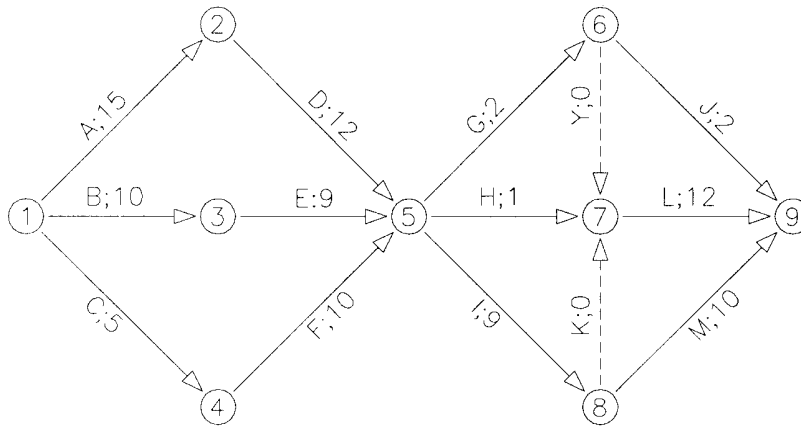


Fig. 1 – Rede PERT

3. Nas bases de dados relacionais a redundância de informação é normalmente indesejável. No entanto, as bases de dados de muitas aplicações comerciais contêm (intencionalmente) campos que podem ser calculados com base noutros campos, ou mesmo campos iguais em tabelas diferentes. Com isto, os programadores destas aplicações procuraram, basicamente, aumentar o desempenho (isto é, reduzir o tempo de processamento utilizado na realização de determinadas tarefas) e/ou simplificar os programas.
De uma forma clara e fundamentada, apresente 2 exemplos (um para cada objectivo) que ilustrem de que forma se pode ganhar desempenho e simplicidade nos programas, não respeitando integralmente o princípio da não redundância de informação, subjacente ao modelo de dados relacional.

V.S.F.F.

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 ao Sistema Gestor de Bases de Dados (SGBD) Access.

Embora os campos do tipo ① _____ e os campos do tipo ② _____ sejam utilizados para armazenar dados com as mesmas características, o primeiro tipo, ao contrário do segundo, exige que seja especificado, à partida, um limite máximo de caracteres.

Um campo numérico definido como *byte* permite uma gama de valores de dimensão ③ _____ à gama de valores de um campo definido como inteiro.

Um campo numérico adequado ao armazenamento de valores monetários deve ser definido como ④ _____.

Para um campo de imagem, o tipo de dados adequado é o tipo 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).

	Ano	Turma	Cód Disciplina	Disciplina
	5	A	21	Educação Visual e Tecnológica
	6	A	21	Educação Visual e Tecnológica
	7	A	22	Educação Visual

Fig. 2 – Formulário de professores, turmas e disciplinas

3. Considere as seguintes tabelas, relativas à gestão de despesas de uma empresa.

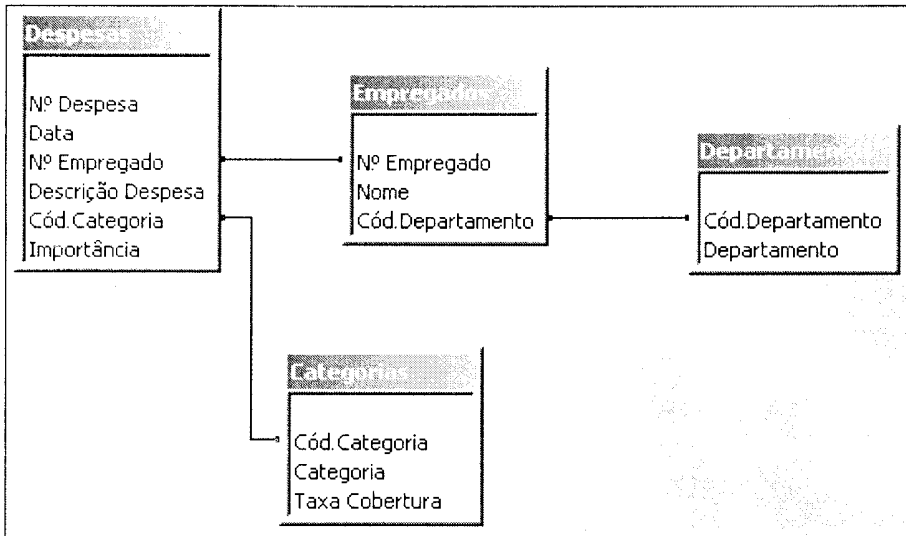


Fig. 3 – Tabelas da base de dados

Nota:

- o «N.º Despesa» é um número sequencial atribuído a cada uma das despesas;
- o campo «Taxa Cobertura» é um número de 0 a 100 correspondente à percentagem da importância da despesa paga ao empregado (coberta) pela empresa;
- os campos «Descrição Despesa», «Nome», «Categoria» e «Departamento» são de texto; o campo «Data» é do tipo data/hora, e os restantes são numéricos.

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. As consultas seguintes (Fig. 4 e Fig. 5) foram construídas mediante a utilização de uma ferramenta gráfica convencional de elaboração de consultas.

3.2.1. Identifique o objectivo da consulta referente à figura 4.

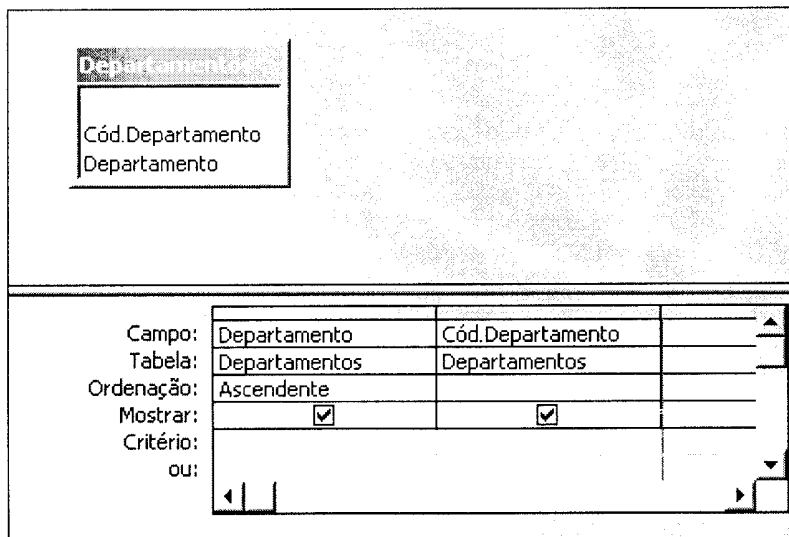


Fig. 4 – Consulta construída com recurso a ferramenta gráfica

V.S.F.F.

3.2.2. Identifique o objectivo da consulta referente à figura 5.

Campo:	Nome		
Tabela:	Empregados		
Ordenação:			
Mostrar:	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Critério:	Como "*Costa"		
ou:			

Fig. 5 – Consulta construída com recurso a ferramenta gráfica

3.3. Tendo em conta que a sintaxe SQL usada utiliza parêntesis rectos para referenciar tabelas e campos cuja designação é formada por mais do que uma palavra, descreva o objectivo de cada uma das instruções SQL seguintes:

3.3.1.

```
SELECT [Cód Departamento],Count(*)
FROM Empregados
GROUP BY [Cód Departamento]
```

3.3.2.

```
SELECT [Cód Categoria],Categoria,[Taxa Cobertura]
FROM Categorias
ORDER BY [Taxa Cobertura] DESC
```

3.3.3.

```
SELECT Sum(Importância)
FROM Despesas
WHERE Year(Data)=2000
```

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

3.4.1. Uma lista com os departamentos da empresa e os nomes dos respectivos empregados, ordenada ascendentemente por departamento e, dentro do departamento, ordenada ascendentemente por nome de empregado.

3.4.2. Para despesas com número no intervalo [150,200], a sua descrição, importância e taxa de cobertura.

3.4.3. A importância total a entregar pela empresa ao funcionário com o número 10, para pagamento das despesas efectuadas no mês 1, do ano 2000.

GRUPO IV

1. Considere o seguinte formulário (*form*) de **Visual Basic** (Fig. 6), que permite visualizar as cartas de um baralho.

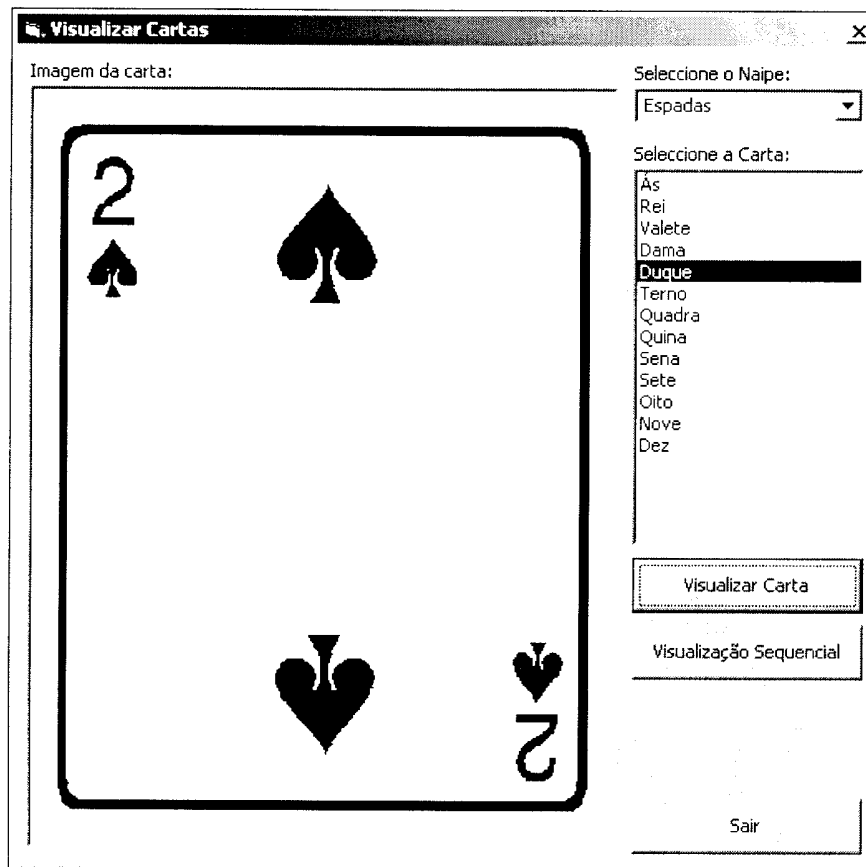


Fig. 6 – Formulário em Visual Basic

Nota: a *combobox* que permite seleccionar o naipe tem o nome *cmbNaipe* e a *listbox* que permite seleccionar a carta tem o nome *lstCarta*.

- 1.1. Para inserir no *form* o texto «Imagem da Carta», «Seleccione o Naipe» e «Seleccione a Carta», qual o tipo de controlo utilizado?
- 1.2. Para que se possa visualizar uma carta é necessário seleccionar previamente o naipe e a carta; caso contrário, deve ser emitida uma das 3 mensagens: «Não foram seleccionados o naipe e a carta», «Não foi seleccionado o naipe» ou «Não foi seleccionada a carta». Codifique o início da sub-rotina seguinte, relativa ao evento *click* do botão «Visualizar Carta», por forma a que seja emitida a mensagem correspondente e interrompido o processamento da sub-rotina, nos casos identificados de má operação.

```
Private Sub cmdVisualizar_Click()
```

```
....
```

```
End Sub
```

- 1.3. Sabendo que o vector *Baralho* (do tipo *string*, global ao *form*, com índices de 0 a 51) foi previamente inicializado (pela ordem dos 4 naipes da *combobox* e, dentro do naipe, pela ordem da *listbox*) com o nome (incluindo a *path*) dos ficheiros que contêm as imagens das 52 cartas, apresente a instrução, a ser incluída na sub-rotina associada ao evento *click* do botão «Visualizar Carta», por forma a que a imagem correspondente seja carregada para o controlo do tipo *imagem imgCarta*.

Nota: assuma que todas as validações foram previamente executadas.

1.4. A visualização sequencial das cartas do baralho é implementada utilizando o *commandbutton* *cmdVisuSeq* e o *timer* *tmrTempoCarta*. O botão de comando permite iniciar/terminar a visualização, e o controlo do tipo *timer* carrega (sequencialmente e temporizadamente) as 52 imagens das cartas.

O *commandbutton* *cmdVisuSeq* reage da seguinte forma, quando é clicado pelo utilizador:

- se o controlo *timer* não está activo, o controlo *timer* é activado e a legenda de *cmdVisuSeq* é alterada para «Parar Visualização Sequencial»;
- se o controlo *timer* está activado, o controlo *timer* é desactivado e a legenda de *cmdVisuSeq* é alterada para «Visualização Sequencial».

Codifique o evento *click* do botão (sub-rotina seguinte), por forma a que o botão tenha o comportamento descrito.

```
Private Sub cmdVisuSeq_Click()
```

```
....
```

```
End Sub
```

1.5. Indique a propriedade que, num controlo do tipo *timer*, permite especificar o intervalo de tempo entre eventos.

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	1.1.	5	40
	1.2.	10	
	1.3.	10	
	1.4.	10	
	1.5.	5	
TOTAL			200