

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

1.ª FASE
2.ª CHAMADA

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 à elaboração de horários para uma escola.

1. Sugira uma estrutura de dados (vector, matriz ou ficheiro) que permita guardar, em memória secundária, o código e o nome das diversas disciplinas. 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 na matriz **HORÁRIO**, previamente preenchida com o horário de um professor, lista as turmas que o professor lecciona. Cada elemento da matriz **HORÁRIO**[I,J] contém a turma (por exemplo, «5B») que o professor lecciona no dia I (o índice 1 corresponde a segunda-feira, o índice 2 corresponde a terça-feira, etc.), ao tempo J (o índice 1 corresponde a 8 h 30 min, o índice 2 corresponde a 9 h 30 min, etc.). Considera-se, ainda, que a matriz **HORÁRIO** está preenchida para 5 dias e 12 tempos e que os elementos da matriz em que o professor tem «furo» estão preenchidos com *string* vazia.

1. [Determinar as turmas que o professor lecciona]
CONT←0
Repita para I=1,2,...,5
 Repita para J=1,2,...,12
 Se HORÁRIO[I,J]≠① _____
 Então Se CONT=0
 Então CONT←1
 TURMAS[② _____]←HORÁRIO[I,J]
 Senão K←1
 Repita enquanto TURMAS[K]≠HORÁRIO[I,J] e K<CONT
 K←K+1
 Se TURMAS[K]≠③ _____
 Então CONT←CONT+1
 TURMAS[CONT]←HORÁRIO[I,J]
2. [Listar as turmas que o professor lecciona]
Repita para ④ _____ =1,2,..., ⑤ _____
 Escreva(TURMAS[K])
3. [Terminar]
Saída

3. Considerando que:
 - a matriz **HORÁRIO** foi previamente preenchida com o horário de um professor;
 - cada elemento da matriz **HORÁRIO**[I,J] contém a turma (por exemplo, «5B») que o professor lecciona no dia I (o índice 1 corresponde a segunda-feira, o índice 2 corresponde a terça-feira, etc.), ao tempo J (o índice 1 corresponde a 8 h 30 min, o índice 2 corresponde a 9 h 30 min, etc.);
 - os elementos da matriz em que o professor tem «furo» estão preenchidos com *string* vazia;
 - a matriz **HORÁRIO** está preenchida para 5 dias e 12 tempos.

Elabore um pequeno algoritmo que determine e imprima o número de horas lectivas do professor.

GRUPO II

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

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

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.

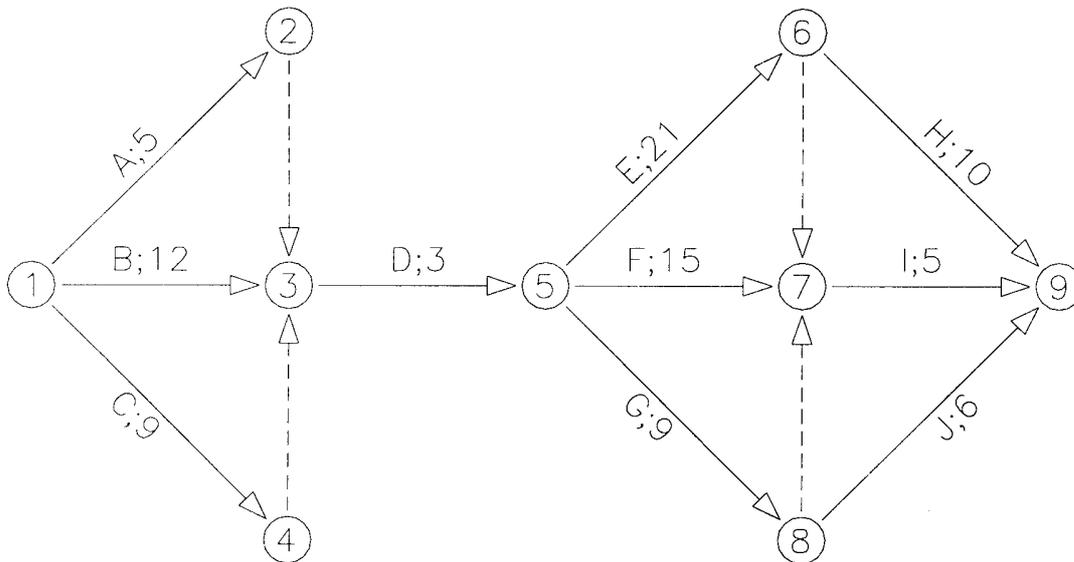


Fig. 1 – Rede PERT

3. Uma das características que as empresas privilegiam ao seleccionar uma aplicação (por exemplo, de gestão) consiste na possibilidade de explorar os dados da aplicação para além do que nela está previsto. Embora seja sempre possível acrescentar módulos aos programas para satisfazer as necessidades dos clientes, é muito mais rápido e barato dar aos utilizadores as ferramentas que lhes permitam obter os resultados pretendidos. Apresente dois aspectos que tornem a exploração de dados armazenados em ficheiros tradicionais mais difícil do que a exploração de dados em bases de dados com formatos conhecidos (por exemplo, Access).

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.

Relativamente à chave primária, pode afirmar-se que:

- é constituída por ① _____ ou ② _____ campos;
- só existe ③ _____ por tabela;
- identifica ④ _____ cada registo da tabela;
- um campo do tipo ⑤ _____ será o candidato ideal a chave primária, pois assume, por natureza, um valor único por registo.

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).

Registo de Ponto

Cód Funcionário: 1
Nome: Luís Campos Santos
Cód Departamento: 100
Departamento: Informática

Funcionários/Marcações

Data	Hora	Tipo Marcação	Descrição Marcação
01-02-2000	9:05:00	1	Entrada
01-02-2000	18:15:00	2	Saída
01-02-2000	13:10:00	3	Saída Almoço
01-02-2000	14:00:00	4	Entrada Almoço

Registo: 5 de 5

Registo: 1 de 1

Fig. 2 – Formulário de registo de ponto de uma empresa

3. Considere as seguintes tabelas, relativas às vendas de uma empresa durante um ano.

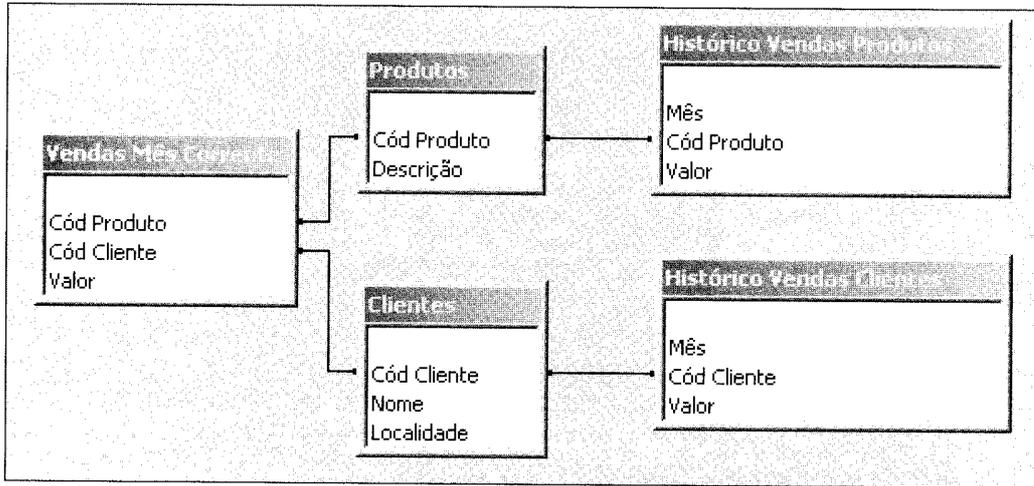


Fig. 3 – Tabelas da base de dados

Nota:

- as tabelas de histórico de vendas, relativas a produtos e a clientes, referem-se às vendas efectuadas nos meses anteriores ao mês corrente;
- os campos «Valor» contêm importâncias em contos;
- os meses são identificados por um número de 1 a 12;
- os campos «Descrição», «Nome» e «Localidade» são de texto, 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

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

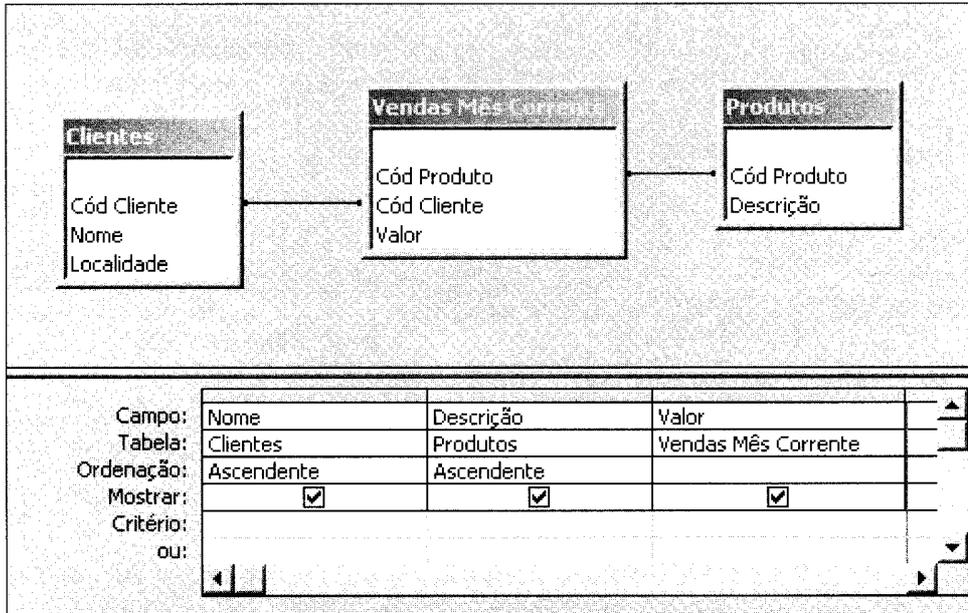


Fig. 5 – Consulta construída com recurso a ferramenta gráfica

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 Valor
FROM [Histórico Vendas Produtos]
WHERE Mês=2 AND [Cód Produto]=1`

3.3.2. `SELECT Sum(Valor)
FROM [Vendas Mês Corrente]
WHERE [Cód Produto]=1`

3.3.3. `SELECT Avg(Valor)
FROM [Histórico Vendas Clientes]
WHERE [Cód Cliente]=1`

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

3.4.1. Relativamente ao mês corrente, o código do produto e o respectivo valor total de vendas.

3.4.2. Para o mês 5, o nome dos diversos produtos vendidos e os respectivos valores das vendas. A lista deverá ser ordenada descendentemente pelos valores das vendas.

3.4.3. O total das vendas acumuladas no ano (excluindo o mês corrente) para todos os clientes, identificados pelo seu nome.

1. Considere o seguinte formulário (form) de **Visual Basic** (Fig. 6), que implementa um pequeno jogo de cartas.

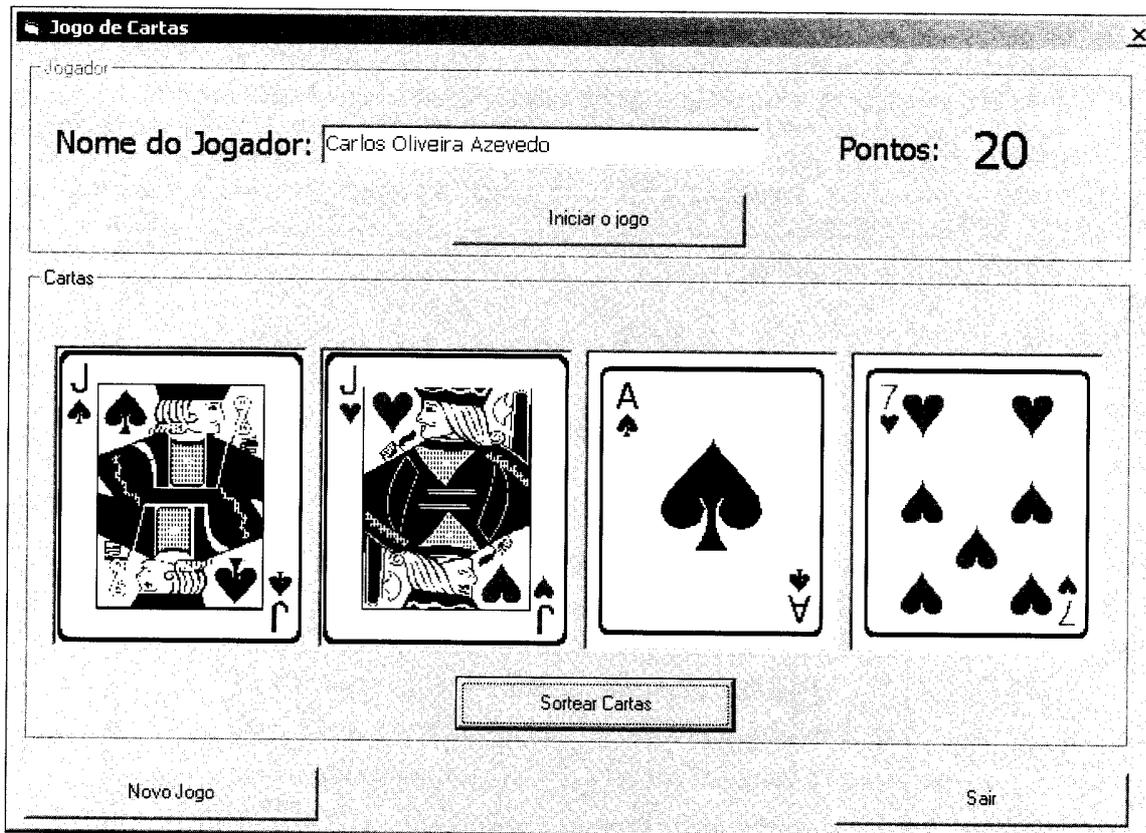


Fig. 6 – Formulário em Visual Basic

- 1.1. Nos controlos do tipo *frame*, qual a propriedade que permite especificar as legendas «Jogador» e «Cartas»?
- 1.2. Quando o jogador clica no botão «Iniciar o jogo» sem ter introduzido o seu nome (conteúdo da *textbox txtNomeJogador* igual a *string* vazia), é emitida a mensagem «Nome do jogador não introduzido», colocado o *focus* na *textbox txtNomeJogador* e abandonada a sub-rotina. Caso contrário, é activada a *frame frmCartas* («Cartas») e desactivada a *frame frmJogador* («Jogador»). Codifique a sub-rotina seguinte, relativa ao evento *click* do botão «Iniciar o jogo».

```
Private Sub cmdIniciar_Click()
    ...
End Sub
```

1.3. Quando o utilizador clica no botão «Sortear Cartas», são escolhidas aleatoriamente 4 cartas do baralho, surgindo as imagens correspondentes no *form*.

Sabendo que:

- o vector de *strings* global ao *form Baralho* foi previamente inicializado com o nome (incluindo a *path*) dos ficheiros que contêm as imagens das cartas;
- a variável *NCartas* (do tipo inteiro global ao *form*) guarda o número de cartas ainda disponíveis no baralho (inicialmente 52);
- as cartas são visualizadas no *form* carregando os ficheiros correspondentes para *imgCarta(0)*, *imgCarta(1)*, *imgCarta(2)* e *imgCarta(3)*;
- quando o utilizador clica no botão e já não existem cartas disponíveis no baralho (*NCartas* igual a zero), é emitida a mensagem «Não existem mais cartas no baralho» e abandonada a sub-rotina;

codifique a sub-rotina seguinte, relativa ao botão «Sortear Cartas», de acordo com as especificações fornecidas.

```
Private Sub cmdSortearCartas_Click()
```

```
...
```

```
End Sub
```

1.4. Para que serve a instrução: *imgCarta(0).Picture=LoadPicture()* ?

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.	20	
	1.4.	5	
TOTAL			200