

(Introdução às)  
Bases de Dados – 2008/09



FACULDADE DE  
CIÊNCIAS E TECNOLOGIA  
UNIVERSIDADE NOVA DE LISBOA

07/Setembro/2009 – 13h – Duração: 3h

Época Especial  
– Sem consulta –

Leia com atenção a informação constante desta página, enquanto espera a indicação do docente para começar a resolução do exame.

Este enunciado é composto por:

- Uma Folha de Rosto (esta)
- Uma Folha de Respostas
- Oito Páginas de Perguntas
- Seis Páginas de Rascunho

O exame é composto por dois grupos de perguntas:

- GRUPO I: Composto por 7 perguntas de resposta curta valendo 8 valores no total.
- GRUPO II: Composto por 16 perguntas de escolha múltipla valendo 12 valores no total.
- Os alunos de Introdução às Bases de Dados não realizam as questões I.6, I.7, II.13, II.14, II.15 e II.16.

GRUPO I:

- A simplicidade e clareza das respostas neste grupo contarão na avaliação. Poderá inclusivamente ter uma cotação de 0 valores numa pergunta se a sua solução for muito mais complicada do que o necessário.
- Todas as perguntas deste grupo têm aproximadamente a mesma cotação.

GRUPO II:

- Cada pergunta tem um número variado de respostas possíveis, onde **apenas uma está correcta**.
- Cada resposta correcta vale 0,75 valores (1 valor para alunos de IBD).
- **Cada resposta incorrecta desconta 0,3 valores (0,4 valores para os alunos de IBD)**. Ou seja, é melhor não responder do que responder errado!
  - Por exemplo, se apenas responder a 6 perguntas e estiverem todas certas, tem 4,5 valores no grupo II. Se, para além das 6 respostas certas, responder erradamente às restantes 10, então terá apenas 1,5 valores no grupo II, dificilmente suficiente para passar no exame.
- A cotação mínima no GRUPO II é de 0 valores.

Preenchimento:

- Todas as respostas deverão ser dadas na folha de respostas fornecida.
- As zonas sombreadas da folha de respostas não deverão ser preenchidas. Para contribuir para a legibilidade da folha de respostas, minimizando rasuras, agradece-se a utilização do espaço de rascunho fornecido (quer ao longo do enunciado, quer no fim do mesmo) antes de escrever a resposta final na folha de respostas.
- No fim de 3 horas de exame os docentes **recolherão apenas a folha de respostas**.

Aconselha-se que veja todas as perguntas do exame antes de começar a sua resolução, para melhor planear a estratégia de resolução. Tem aproximadamente 7 minutos e 45 segundos por pergunta pelo que não deve demorar demasiado tempo em cada uma.

Boa Sorte!

m

## GRUPO I

Considere uma base de dados para registar informação sobre mesas de voto e eleitores das eleições autárquicas:

```
Concelhos ( { IdC, NomeC, NomeD } )
Freguesias ( { IdC, NomeF, PopF } )
Mesas ( { IdC, NomeF, NrM, MinE, MaxE } )
Eleitores ( { IdC, NomeF, NrE, NomeE,
             BiE, DtNscE, DtObE } )
```

Os concelhos do país encontram-se registados na tabela *Concelhos*, com um identificador (*IdC*), nome do concelho (*NomeC*) e nome do distrito (*NomeD*). Cada freguesia tem uma chave primária formada pelo identificador do seu concelho e pelo nome da freguesia (*NomeF*); regista-se ainda na tabela *Freguesias* a população da respectiva freguesia (*PopF*), ou seja, o número total de habitantes dessa freguesia eleitores ou não. A tabela *Mesas* regista a informação sobre as mesas de votos. Em cada freguesia existem 1 ou mais mesas de voto, numeradas sequencialmente dentro de cada freguesia a partir de 1 (coluna *NrM*), podendo exercer o direito de voto nessa mesa de voto os eleitores recenseados nessa freguesia com número de eleitor entre *MinE* e *MaxE* (inclusivo). A tabela *Eleitores* regista os eleitores alguma vez recenseados na freguesia identificada por (*IdC, NomeF*) em que *NrE* é o número sequencial de eleitor atribuído por cada freguesia no acto de recenseamento; são ainda armazenados o nome do eleitor, o seu BI, a data de nascimento do eleitor (*DtNscE*) e a possível data de óbito (*DtObE*). Nesta base de dados só a coluna *DtObE* pode conter valores nulos, significando que o eleitor se encontra vivo e logo pode votar.

**I.1.** Crie em SQL a vista *cadernos\_eleitorais* (*IdC, NomeF, NrM, NrE, NomeE, BiE*) que permita listar os cadernos eleitorais de todas as mesas de voto do país, onde só devem constar eleitores vivos. Uma entrada desta vista significa que o eleitor com o número *NrE* da freguesia identificada por *IdC, NomeF* deverá votar na mesa *NrM* dessa freguesia. Os cadernos devem estar ordenados por *IdC, NomeF, NrM* e *NrE*.

**I.2.** Apresente uma consulta em SQL que permita saber os nomes dos concelhos do Distrito de Lisboa cuja população total (soma da população das suas freguesias) é igual ou superior a 50000 habitantes.

**I.3.** Adicione uma restrição garantindo que não há eleitores com o mesmo número de bilhete de identidade.

Apresente consultas em Álgebra Relacional correspondentes a cada uma das expressões das seguintes duas alíneas:

**I.4.** Quais os nomes das freguesias do concelho do Porto que têm um eleitor com o número 157 nascido após 1979?

**I.5.** Obtenha as mesas de voto que estão imediatamente precedidas por outra mesa de voto, devendo ser listado *IdC, NomeF, NrM*. Por exemplo, se uma instância da base de dados contiver apenas as mesas 1,2 e 4 da freguesia identificada por (15, 'Carnide') deverá constar somente o tuplo (15, 'Carnide', 2) na relação resultado, pois não existe nem mesa 0 nem mesa 3 nessa freguesia.

XX

Considere o seguinte XML DTD, definido para registar em XML a constituição das listas a uma dada freguesia:

```
<!DOCTYPE Listas [
  <!ELEMENT Listas (Lista+)>
  <!ATTLIST Listas IdC CDATA #REQUIRED NomeF CDATA #REQUIRED>
  <!ELEMENT Lista (NomeL,CandidatosL)>
  <!ATTLIST Lista SiglaL ID #REQUIRED>
  <!ELEMENT NomeL (#PCDATA)>
  <!ELEMENT CandidatosL (Eleitor, Eleitor, Eleitor+)>
  <!ELEMENT Eleitor (NomeE, BI)>
  <!ELEMENT NomeE (#PCDATA)>
  <!ELEMENT BI EMPTY>
  <!ATTLIST BI NrBI ID #REQUIRED DataValidade CDATA #REQUIRED
            DataEmissao CDATA #IMPLIED> ]>
```

**I.6.** Apresente o menor documento XML que use esta DTD e que contenha informação relativamente à freguesia com do concelho com identificador 1 e nome de freguesia "Lumiar". Os nomes das listas e dos candidatos devem ser não vazios. Recorde ainda que em XML, os identificadores devem sempre começar por uma letra.

**I.7.** Escreva uma expressão XPath que devolva as siglas das listas que têm um candidato chamado 'João Manuel'.



II.3. Considere as seguintes tabelas:

```
CREATE TABLE R(a INT, b CHAR, c INT NOT NULL,  
PRIMARY KEY (a,b) );  
CREATE TABLE S(a INT, b CHAR, d INT NOT NULL,  
FOREIGN KEY (a,b) REFERENCES R);
```

Assuma que as tabelas contêm os seguintes dados:

R:

A	B	C
1	a	3
1	b	2
2	a	3
2	b	3

S:

A	B	D
1	a	4
1	a	1
2	a	7
2	a	8

Considere os dois comandos SQL:

- I. DELETE FROM R WHERE B='a';
- II. DELETE FROM S WHERE B='a';

Qual das seguintes afirmações é verdadeira?

- (A) Nenhum dos comandos remove linhas das tabelas.
- (B) Só o comando I remove linhas de tabelas.
- (C) Só o comando II remove linhas de tabelas.
- (D) Ambos os comandos removem linhas de tabelas.



II.4. Considere a tabela P(a,b,c) criada e instanciada com os seguinte comandos:

```
CREATE TABLE P(a INT, b INT, c INT NOT NULL, PRIMARY KEY (a,b));  
INSERT INTO P VALUES (5, 2, 7);
```

Posteriormente foi implementado o seguinte trigger

```
CREATE TRIGGER t1  
BEFORE INSERT ON P FOR EACH ROW  
BEGIN  
IF :NEW.a > :NEW.b  
THEN INSERT INTO P VALUES (:NEW.a-1, :NEW.b+1, :NEW.c);  
END IF;  
END;
```

Considere os três comandos SQL:

- I. INSERT INTO P VALUES (6, 1, 8);
- II. INSERT INTO P VALUES (4, 3, 8);
- III. UPDATE P SET P.a = P.a - 2, P.b = P.b + 2, P.c = 8;

Quais dos comandos anteriores fazem a linha (3,4,8) pertencer à tabela P?

- (R) Nenhum
- (S) Apenas I
- (T) Apenas II
- (U) Apenas III
- (V) I e II
- (W) I e III
- (X) II e III
- (Y) Todos

II.5. Considere as tabelas R(A,B) e S(B,C) com as seguintes linhas:

R:

A	B
1	1
1	3
2	3

S:

B	C
1	2
1	3
2	4

Qual das seguintes tabelas é o resultado da consulta SQL abaixo, ignorando a ordem das linhas?

```
SELECT A+B+C AS D  
FROM R NATURAL LEFT OUTER JOIN S;
```

(J)

D
4
5
NULL

(K)

D
4
5
NULL
NULL

(L)

D
4
5

(M)

D
4
5
4
5

**Espaço de rascunho:**

II.6. Considere as seguintes tabelas:

```
CREATE TABLE R(a INT PRIMARY KEY, b INT NOT NULL);  
CREATE TABLE S(a INT PRIMARY KEY, c INT NOT NULL,  
FOREIGN KEY (a) REFERENCES R ON DELETE CASCADE);
```

Suponha ainda que as tabelas R e S contêm  $m$  e  $n$  tuplos respectivamente. Seja  $t$  o número de tuplos no resultado da seguinte consulta:

```
SELECT * FROM R NATURAL INNER JOIN S;
```

Qual é o valor exacto de  $t$ ?

(J)  $t = m.$

(K)  $t = n.$

(L)  $t = m+n.$

(M)  $t = m*n.$

**Espaço de rascunho:**







