

**LABORATÓRIO DE INTELIGÊNCIA ARTIFICIAL E DE COMPUTADORES**  
**Mestrado em Inteligência Artificial e Computação**  
**Sistemas de Bases de Dados**  
**Exame, 16 de Abril de 1998**  
**DURAÇÃO MÁXIMA 2 horas, com consulta**

**Problema 1: Modelização de dados (6 valores • 40 minutos)**

Os Bombeiros Voluntários de Atouguia da Baleia desejam passar a registar as utilizações das suas viaturas e para isso encomendaram um sistema de informação.

Para cada viatura interessa registar a matrícula, a marca, a operacionalidade (a funcionar ou inoperacional) e o tipo de viatura (auto-tanque, ambulância ou carro de comando).

Para todas as saídas das viaturas devem ser registadas a data e hora de saída, os quilómetros iniciais e finais, o bombeiro que a conduziu e a ocorrência a que diz respeito. Uma viatura só sai uma vez para cada ocorrência e só bombeiros habilitados a podem conduzir.

As ocorrências são identificadas por um código numérico sequencial dentro de cada tipo de ocorrência e interessa registar a data e hora em que foi comunicada ao comando dos bombeiros, o contacto da pessoa que fez a comunicação da ocorrência e o tipo de ocorrência (fogo, acidente ou salvamento).

Os bombeiros são identificados pelo seu número interno e interessa registar o seu nome, morada, data de nascimento e posto (comandante, chefe, sub-chefe, sargento ou praça) e se pode, ou não, conduzir viaturas da corporação. Só estes bombeiros podem conduzir viaturas.

- 1.1** (4.0) Obtenha um modelo Entidade/Associação representativo da informação acima descrita. Não se esqueça de indicar a multiplicidade das associações; indique também as chaves, os atributos e pelo menos duas restrições adicionais.
- 1.2** (2.0) Converta o diagrama obtido em 1.1 para um esquema relacional, seguindo a metodologia exposta nas aulas. Justifique devidamente as opções tomadas; utilize uma notação abreviada da forma  $R1(\underline{A1}, \underline{A2}, A3, \dots, An)$ , em que  $R1$  é o nome de uma relação,  $A1, \dots, An$  são nomes de atributos e os atributos sublinhados constituem a chave primária; indique à parte as chaves alternativas e as chaves externas.

**Problema 2: Dependências funcionais e normalização (2 valores • 10 minutos)**

Considere uma relação  $R(A, B, C, D)$ . Para o conjunto de dependências funcionais  $F$  a seguir indicado, determine as dependências não triviais que decorrem de  $F$ , as chaves da relação, uma superchave da relação que não é chave, as eventuais violações da forma normal de Boyce-Codd (BCNF) e no caso de existir violação, uma decomposição de  $R$  em duas ou mais relações na BCNF; prove que não há perda de dependências funcionais como resultado da decomposição.

$$F = \{AB \rightarrow C, C \rightarrow D\}$$

**Problema 3: Interrogação e manipulação de dados em SQL (3,5 valores • 20 minutos)**

Suponha que tem uma base de dados com informação relativa a uma árvore genealógica, com as seguintes tabelas:

Pessoa (bi, nome, data\_nascimento, bi\_mãe, bi\_pai, sexo)

Casamento (bi\_marido, bi\_mulher, data\_início, data\_fim, número\_filhos)

Para além das chaves primárias indicadas (a sublinhado), suponha que se verificam as seguintes restrições de integridade:

R1: O sexo do marido (mulher) é 'M' ('F')

R2: A data do divórcio (data\_fim) é maior que a data do casamento (data\_início)

R3: A idade de um filho é sempre menor que a dos pais

R4:  $\forall_{c \in \text{Casamento}}, c.\text{número\_filhos} = \#[\sigma_{\text{pai} = c.\text{bi\_marido}, \text{mãe} = c.\text{bi\_mulher}}(\text{Pessoa})]$

**(continua...)**

Formule em SQL as seguintes questões:

- 3.1 Mostrar os nomes dos filhos da pessoa com número de bilhete de indentidade bi=1234567, por ordem alfabética.
- 3.2 Mostrar, os pares de nomes de irmãos, sem repetições de pares.
- 3.3 Mostrar os números de BI das pessoas com pais conhecidos e que nunca estiveram casados.
- 3.4 Actualizar o atributo número\_filhos da tabela “Casamento” recalculando-o com base na informação existente na tabela “Pessoa” por forma a satisfazer a restrição R4.
- 3.5 Eliminar os casamentos que violam a restrição R2.
- 3.6 Inserir na tabela “Pessoa”, uma ocorrência com número de BI 1231234, chamado João, com mãe 1234567 e pai desconhecido.

**Problema 4: Definição de dados em SQL (3,5 valores • 20 minutos)**

Considere de novo a base de dados do Problema 3.

- 4.1 Escreva comandos em SQL para criar as duas tabelas indicadas, com as chaves primárias indicadas, não precisando de contemplar as restrições R1 a R4.
- 4.2 Escreva comandos em SQL para alterar a definição das tabelas criadas em 4.1, por forma a impôr as restrições R1 e R2.
- 4.3 Escreva uma asserção em SQL para impôr a restrição R3.
- 4.4 Escreva um ou mais gatilhos em SQL para impôr a restrição R4 de forma incremental: ao inserir, eliminar ou actualizar uma linha da tabela “Pessoa”, deve ser actualizada, se existir, a instância respectiva da tabela “Casamentos”.

**Problema 5: Linguagens de programação de bases de dados (5 valores • 30 minutos)**

- 5.1 Sistemas de tipos possuem uma função dupla de protecção e de modelização. Diga em que medida tipos de dados abstractos, polimorfismos e reflexão linguística aumentam a capacidade de modelização de linguagens de programação fortemente tipadas.
- 5.2 Interprete a seguinte regra de tipos e refira a sua aplicação prática:

$$\frac{\Gamma \vdash s_1 \leq t_1 \quad \dots \quad \Gamma \vdash s_n \leq t_n}{\Gamma \vdash [a_1:s_1, \dots, a_n:s_n, \dots, a_m:s_m] \leq [a_1:t_1, \dots, a_n:t_n]}$$

- 5.3 As linguagens de programação de bases de dados são desenhadas para tratar grandes quantidades de informação que se mantêm activas por muito tempo e usualmente são acedidas simultaneamente por vários utilizadores. Diga quais as extensões ao modelo relacional baseadas em atributos contendo valores estruturados que foram propostas e em que consistem.
- 5.4 Analise à luz dos modelos de equivalência de tipos por nome e estrutural, a equivalência de expressões dos seguintes tipos:
 

```

type Animal is record(
    Age: Integer;
    Weight: Real)
type Vehicle is record(
    Age: Integer;
    Weight: Real)

```
- 5.5 Diga porque é que equivalência de tipos por nome não se adapta bem ao desenvolvimento incremental de aplicações.

**(Fim.)**