

# Prova de Fundamentos de Bancos de Dados

1ª Prova

Prof. Carlos A. Heuser

Abril de 2010

**Nome do aluno:**

**Prova sem consulta** – duas horas de duração

1. (*Peso 2*)

Deseja-se projetar uma base de dados para controlar a ocupação dos laboratórios do Instituto de Informática. Nesta base de dados, estão armazenadas informações sobre os laboratórios, os pontos de trabalho neles existentes e as pessoas que os ocupam.

Um laboratório é identificado pelo número do prédio e da sala em que ele se encontra. O mesmo número de sala pode aparecer em diferentes prédios. Para cada laboratório é necessário conhecer sua descrição e os pontos de trabalho nele contidos.

Os pontos de trabalho de um laboratório são numerados de um em diante. Cada ponto de trabalho é descrito por um texto livre (algo como “PC Core2 Duo 4GB/320GB/19”). Para cada ponto de trabalho é necessário conhecer os usuários deste ponto de trabalho, juntamente com o número de horas semanais que cada usuário trabalha no ponto em questão. Observe que um determinado usuário pode utilizar diferentes pontos de trabalho. Cada usuário é identificado pelo seu número de cartão na UFRGS. Além disso, é necessário conhecer o nome do usuário.

Projete uma base de dados relacional para armazenar os dados acima sem redundância de dados. Enumere as tabelas, suas colunas, as chaves primárias e as chaves estrangeiras. Não devem ser criadas colunas artificiais, além das apresentadas no enunciado. Apresente o esquema na notação textual ou diagramática vistas em aula.

**Solução:**

```
Laboratorio (num_pred, num_sala, descricao_lab)
Ponto (num_pred, num_sala, num_ponto, descricao_ponto)
        (num_pred, num_sala) referencia Laboratorio
PontoUsuario (num_pred, num_sala, num_ponto, num_cartao,
               horas_sem)
        (num_pred, num_sala, num_ponto) referencia Ponto
        (num_cartao) referencia Usuario
Usuario (num_cartao, nome_usuario)
```

2. Considere a seguinte base de dados, usada pelo Instituto de Informática para conhecer seus grupos de pesquisa e a constituição destes grupos.

*/\* tabela de grupos de pesquisa \*/*

Grupo (cod\_grupo, nome\_grupo)

*/\* tabela com dados das pessoas \*/*

Pessoa (no\_cartao, nome, codigo\_papel);  
    (codigo\_papel) references Papel

*/\* tabela com o papel (docente, aluno de mestrado, bolsista, ... que cada pessoa \*/*

Papel (codigo\_papel, descricao\_papel);

*/\* tabela com os orientadores de cada aluno \*/*

Orientacao (no\_cartao\_orientador, no\_cartao\_orientando)  
    (no\_cartao\_orientador) references Pessoa  
    (no\_cartao\_orientando) references Pessoa

*/\* tabela com os participantes dos grupos de pesquisa \*/*

Participacao (no\_cartao, cod\_grupo)  
    (no\_cartao) references Pessoa  
    (cod\_grupo) references Grupo

Sobre esta base de dados, resolver as consultas que seguem usando *álgebra relacional*. Não usar mais tabelas que o estritamente necessário.

a) (*Peso 1,33...*)

Obter os nomes dos alunos de mestrado orientados pelo professor de nome 'José Silva'. Sabe-se que uma pessoa é aluno de mestrado quanto tem o papel com descrição 'Aluno de mestrado'.

i. Resolver usando produto cartesiano.

**Solução:**

```
 $\pi$  Aluno.nome
  ( $\sigma$  Professor.no_cartao =
    Orientacao.no_cartao_orientador AND
    Aluno.no_cartao =
      Orientacao.no_cartao_orientando AND
    Papel.cod_papel = Aluno.CodPapel AND
    Papel.descricao_papel = 'Aluno de mestrado' AND
    Professor.nome = 'José Silva'
    (( $\rho$  Professor (Pessoa))  $\times$ 
     ( $\rho$  Aluno (Pessoa))  $\times$ 
     Orientacao  $\times$ 
     Papel
    )
  )
```

ii. Resolver usando os operadores de junções. Qualquer combinação de operadores de junção pode ser usada (junção natural, equi-junção e theta-junção).

**Solução:**

```
 $\pi$  Aluno.nome
  ( $\sigma$ 
    Papel.descricao_papel = 'Aluno de mestrado' AND
    Professor.nome = 'José Silva'
    (( $\rho$  Professor (Pessoa)
       $\bowtie$  Professor.no_cartao =
        Orientacao.no_cartao_orientador
        Orientacao
      )
       $\bowtie$  Aluno.no_cartao =
        Orientacao.no_cartao_orientando AND
      ( $\rho$  Aluno (Pessoa)  $\bowtie$  Papel)
    )
  )
```

b) (*Peso 1,33...*)

Obter os códigos e nomes dos grupos de pesquisa que estão sem participantes.

**Solução:**

```
Grupo
—
(π cod_grupo, nome_grupo
  (Grupo ⋈ Participacao)
)
```

c) (*Peso 1,33...*)

Obter os nomes das pessoas que participaram em todos grupos de pesquisa cadastrados na base de dados

**Solução:**

```
π Pessoa.nome
(  (Participacao
    ÷
    (π cod_grupo (Grupo))
  )
 ⋈
 Pessoa
)
```

d) (*Peso 1,33...*)

Obter **uma única** tabela contendo as seguintes colunas:

- i. Código e descrição de cada grupo de pesquisa;
- ii. código e nome de cada professor que atua no grupo (se não houver professores no grupo, estas colunas devem aparecer em branco). Uma pessoa é professor quando a descrição de seu papel é 'Professor'.

**Solução:**

```
 $\pi$  Grupo.cod_grupo,  
   Grupo.nome_grupo,  
   Pessoa.no_cartao,  
   Pessoa.nome  
   (Grupo  
      $\bowtie$   
     ( (  $\sigma$  descricao_papel = "Professor"  
         (Participacao  $\bowtie$   
           Pessoa  $\bowtie$   
             Papel  
             )  
         )  
     )  
   )  
 )
```

3. (Peso 1,33...)

Sobre a base de dados da questão precedente (Questão 2), expresse a seguinte consulta em SQL, *sem* o uso da sintaxe para junções explícitas, *nem* sub-consultas (ainda não vistas em aula). Na consulta, devem aparecer apenas as tabelas necessárias.

Obter o nome de cada orientador, seguido do nome de cada um de seus orientandos. Somente orientadores e orientandos aparecem no resultado. Outras pessoas não aparecem.

**Solução:**

```
SELECT Otdor.nome,  
       Otndo.nome,  
       A2.placa  
FROM Orientacao AS O,  
     Pessoa AS Otdor,  
     Pessoa AS Otndo  
WHERE  
       Otdor.no_cartao = O.no_cartao_orientador AND  
       Otndo.no_cartao = O.no_cartao_orientando
```

4. (Peso 1,33...)

Considere a seguinte consulta em SQL:

```
SELECT Pessoa.nome
FROM Papel,
      Pessoa,
      Participacao
WHERE
      Papel.cod_papel = Pessoa.cod_papel AND
      Pessoa.num_cartao = Participacao.num_cartao AND
      Papel.descricao_papel = 'Bolsista' AND
      Participacao.cod_grupo = 'SI03'
```

Mostre a consulta equivalente em álgebra relacional, depois, mostre a representação da consulta em forma de árvore e após, mostre cada um dos passos da otimização algébrica.

**Solução:**

Consulta em álgebra relacional (a ordem dos produtos cartesianos foi arbitrada; outras ordens são equivalente e levam a uma consulta diferente com o mesmo resultado):

```
 $\pi$  Pessoa.nome
 $\sigma$ 
      Papel.cod_papel = Pessoa.cod_papel AND
      Pessoa.num_cartao = Participacao.num_cartao AND
      Papel.descricao_papel = 'Bolsista' AND
      Participacao.cod_grupo = 'SI03'
      ( Papel
        ×
        (Pessoa
          ×
          Participacao
        )
      )
```