



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**  
**Disciplina Banco de Dados**  
**AD2 2º semestre de 2011.**

Nome: \_\_\_\_\_

Observações:

**1. Prova COM consulta.**

**Atenção:** Como a avaliação à distância é individual, caso seja constatado que provas de alunos distintos são cópias umas das outras, independentemente de qualquer motivo, a todas será atribuída a nota ZERO. As soluções para as questões podem sim, ser buscadas por grupos de alunos, mas a redação final de cada prova tem que ser individual.

ADs enviadas pelo correio devem ser postadas cinco dias antes da data final de entrega estabelecida no calendário de entrega de ADs.

**Questão 1 (1 ponto – 0,2 cada item).** Considere o seguinte esquema relacional, onde as chaves primárias estão sublinhadas.

Funcionario ( <u>fid</u> : integer, fnome: string, salario:real, did:integer) did referencia Departamento Departamento ( <u>did</u> : integer, dnome: string)
---

Sobre esta base de dados, resolver as consultas a seguir usando álgebra relacional.

**ATENÇÃO:** Não usar mais tabelas que o estritamente necessário.

- (a) Quais os nomes dos empregados que ganham um salário menor que 2.000,00.  
 $\pi_{fnome} (\sigma_{salario < 2000} Funcionario)$
- (b) Quais os nomes dos departamentos que possuem funcionários que ganham salário até 1.500,00.  
 $\pi_{dnome} (\sigma_{salario < 1500} Funcionario \bowtie Departamento)$
- (c) Quais os nomes dos empregados que trabalham no departamento "D3".  
 $\pi_{fnome} (Funcionario \bowtie \sigma_{dnome = "D3"} Departamento)$

- (d) Quais os nomes de empregado que trabalham no departamento “Contabilidade”.

$\pi_{fnome} (\text{Funcionario} \bowtie \sigma_{dnome = \text{“Contabilidade”}} \text{Departamento})$

- (e) Quais os nomes dos departamentos não possuem empregados.

$\pi_{dnome} (\text{Departamento}) - \pi_{dnome} (\text{Funcionario} \bowtie \text{Departamento})$

**Questão 2 (1 ponto – 0,2 cada item).** Sobre a base de dados da questão anterior, resolver as consultas utilizando SQL. Não usar mais tabelas que o estritamente necessário.

- (a) Quais os nomes dos empregados que ganham um salário menor que 2.000,00.

**SELECT fnome FROM Funcionario  
WHERE salario < 2000;**

- (b) Quais os nomes dos departamentos que possuem empregados que ganham salário até 1.500,00.

**SELECT dnome FROM Funcionario f, Departamento d  
WHERE f.did = d.did  
AND f.salario < 1500;**

- (c) Quais os nomes dos empregados que trabalham no departamento “D3”.

**SELECT fnome FROM Funcionario f, Departamento d  
WHERE f.did = d.did  
AND d.dnome = “D3”;**

- (d) Qual o nome do departamento em que trabalha o empregado “Daniel de Oliveira”.

**SELECT dnome FROM Funcionario f, Departamento d  
WHERE f.did = d.did  
AND f.fnome = “Daniel de Oliveira”;**

- (e) Quais os nomes dos departamentos não possuem empregados.

**SELECT dnome from  
DEPARTAMENTO  
WHERE did NOT IN (SELECT did FROM DEPARTAMENTO d, EMPREGADO e  
WHERE d.did = e.did);**

**Questão 3 (2 pontos – 0,2 cada item).** Considere o seguinte esquema relacional:

Cliente (cid: integer, cnome: string, end: string)  
 Produto (pid: integer, pname: string, tid: integer, preço: real)  
     tid referencia Tipo  
 Compra (cid: integer, pid: integer, quantidade: integer)  
     cid referencia Cliente  
     pid referencia Produto  
 Tipo (tid: integer, tnome: string)

No esquema acima, as chaves primárias estão sublinhadas. A tabela de Compra lista a quantidade de itens de um mesmo produto comprados por um cliente. Apresente, para cada consulta a seguir, as expressões em álgebra relacional correspondentes às consultas.

1. Obtenha o pid do produto de nome “produto A”.

$\Pi_{pid} (\sigma_{pname="Produto A"} Produto)$

2. Obtenha o nome dos produtos que foram comprados pelo menos uma vez.

$\pi_{pname} (Produto \bowtie Compra)$

3. Obtenha o nome dos clientes que compraram produtos do tipo “A” e o nome do produto comprado.

$\pi_{cnome, pname} (Cliente \bowtie Compra \bowtie Produto \bowtie \sigma_{tnome="A"} Tipo)$

4. Obtenha o nome dos produtos do tipo “C” que tenham preço superior a 150.

$\pi_{pname} ((\sigma_{tnome="C"} Tipo) \bowtie (\sigma_{preço > 150} Produto))$

5. Obtenha o pid dos produtos que são dos tipos “A” ou “B”.

$\pi_{pid} ((\sigma_{tnome="A" \vee tnome="B"} Tipo) \bowtie Produto)$

6. Obtenha o nome dos clientes que compraram algum produto do tipo “C” em quantidade superior a 10 unidades.

$\pi_{cnome} ((\sigma_{quantidade > 10} Compra \bowtie (Produto \bowtie \sigma_{tnome="C"} Tipo)) \bowtie Cliente)$

7. Obtenha o cid dos clientes que compraram mais de 20 unidades de um item ou que compraram itens do tipo “B”.

$\rho(R1, \pi_{cid}(\sigma_{quantidade > 20} Compra))$

$\rho(R2, \pi_{cid}(\pi_{pid}((\pi_{tid} \sigma_{tnome="C"} Tipo) \bowtie Produto) \bowtie Compra))$

$R1 \cup R2$

8. Obtenha o nome dos clientes que compraram o produto “Produto Z” e o produto “Produto K” em quantidade maior a 10 unidades cada.

$\rho(R1, \pi_{cid}((\pi_{pid} \sigma_{pname="Produto Z"} Produto) \sigma_{quantidade > 10} Compra))$

$\rho(R1, \pi_{cid}((\pi_{pid} \sigma_{pname="Produto K"} Produto) \sigma_{quantidade > 10} Compra))$

$\rho(R3, R1 \cap R2)$

$\pi_{cnome}(Cliente \bowtie R3)$

9. Obtenha o nome dos clientes que compraram algum produto de preço abaixo de 100.

$\pi_{cnome} ((\sigma_{preço < 100} Produto) \bowtie Compra) \bowtie Cliente)$

10. Obtenha o nome dos tipos dos produtos comprados pelo cliente “Daniel” e que não foram comprados pelo cliente “Pedro” em quantidade superior a 50 itens.

$\rho(R1, \pi_{tnome} (((\sigma_{cnome = \text{“Daniel”}} Cliente) \bowtie Compra) \bowtie Produto) \bowtie Tipo))$

$\rho(R2, \pi_{tnome} (((\sigma_{cnome = \text{“Pedro”}} Cliente) \bowtie \sigma_{quantidade > 50} Compra) \bowtie Produto) \bowtie Tipo))$

$R1 - R2$

**Questão 4 (3 pontos).** Considere a seguinte base de dados, usada por uma empresa de computadores (Dell, HP, por exemplo) e que disponibiliza manutenção de computadores e *upgrades*. As chaves primárias estão sublinhadas.

```
CLIENTE (cpf, nome_cli)

COMPUTADOR (no_serie, modelo, cpf);

(cpf) references CLIENTE

/* tabela com os upgrades periódicos programados e realizados – para cada computador, a
empresa cadastra todos os upgrades programados. */

UPGRADE_REVISAO (no_serie, data_programada, data_ultimo_upgrade,
data_executada)

(no_serie) references COMPUTADOR

PEÇA_UPGRADE_REVISAO (no_serie, data_programada, cod_peça, quantidade)

(no_serie, data_programada) references UPGRADE_REVISAO

(cod_peça) references PEÇA

PEÇA (cod_peça, descricao_peça)
```

Sobre esta base de dados, resolver as consultas a seguir usando álgebra relacional.

**ATENÇÃO:** Não usar mais tabelas que o estritamente necessário.

- (a) Faça uma consulta que retorna os nomes dos clientes que possuem computador do modelo Vaio NBX [0,5 ponto].

$\pi_{nomecli, (Cliente \bowtie (\sigma_{modelo = \text{“Vaio NBX”}} Computador))}$

(b) Faça uma consulta que retorna os nomes dos clientes e o modelo do computador, cujo computador teve um upgrade executado na data de 2011-02-20 [0,5 ponto].

$\pi_{\text{nomecli, modelo, (Cliente} \bowtie \text{Computador (} \bowtie \sigma_{\text{data\_executada} = \text{"2011-02-20"} \text{ Upgrade\_Revisao))}}$

(c) Faça uma consulta que retorna a descrição das peças que nunca foram usadas em upgrades [1,0 ponto].

$\pi_{\text{descricao\_peca (Peça} \bowtie (\pi_{\text{cod\_peça (Peça)} - \pi_{\text{cod\_peça (Peça\_Upgrade\_Revisao))})}$

(d) Obter o número de série dos computadores com a data programada do próximo upgrade dos computadores que, em uma determinada revisão, utilizaram mais do que 1 peça com descrição = "RAM DDR 2" [1,0 ponto].

$\pi_{\text{no\_serie}}$

$(\sigma_{\text{descricao\_peca} = \text{"RAM DDR 2"} \text{ AND quantidade} > 1}$

$(\text{UPGRADE\_REVISAO} \bowtie \text{PECA\_UPGRADE\_REVISAO} \bowtie \text{PECA})$

)

**Questão 5 (1 ponto – 0.2 cada item).** Sobre a base de dados do exercício anterior, resolver as consultas a seguir usando SQL.

**ATENÇÃO:** Não usar mais tabelas que o estritamente necessário.

(a) Escreva uma instrução SQL para inserir um computador de número de série M4N68T-M, modelo igual a Acer Aspire , e CPF igual a 1010. [0.2 ponto]

**INSERT INTO COMPUTADOR (no\_serie, modelo, cpf)  
VALUES ("M4N68T-M", "Acer Aspire", "1010")**

(b) Escreva uma instrução SQL para excluir a tabela Cliente. [0.2 ponto]

**DROP TABLE Cliente**

(c) Faça uma consulta SQL que retorna o nome e cpf de clientes que possuem algum computador cadastrado. O resultado deve estar ordenado pelo nome do cliente. [0.2 ponto]

**SELECT c.nome, c.cpf  
FROM Cliente c, Computador co  
WHERE c.cpf = co.cpf  
ORDER BY c.nome\_cli**

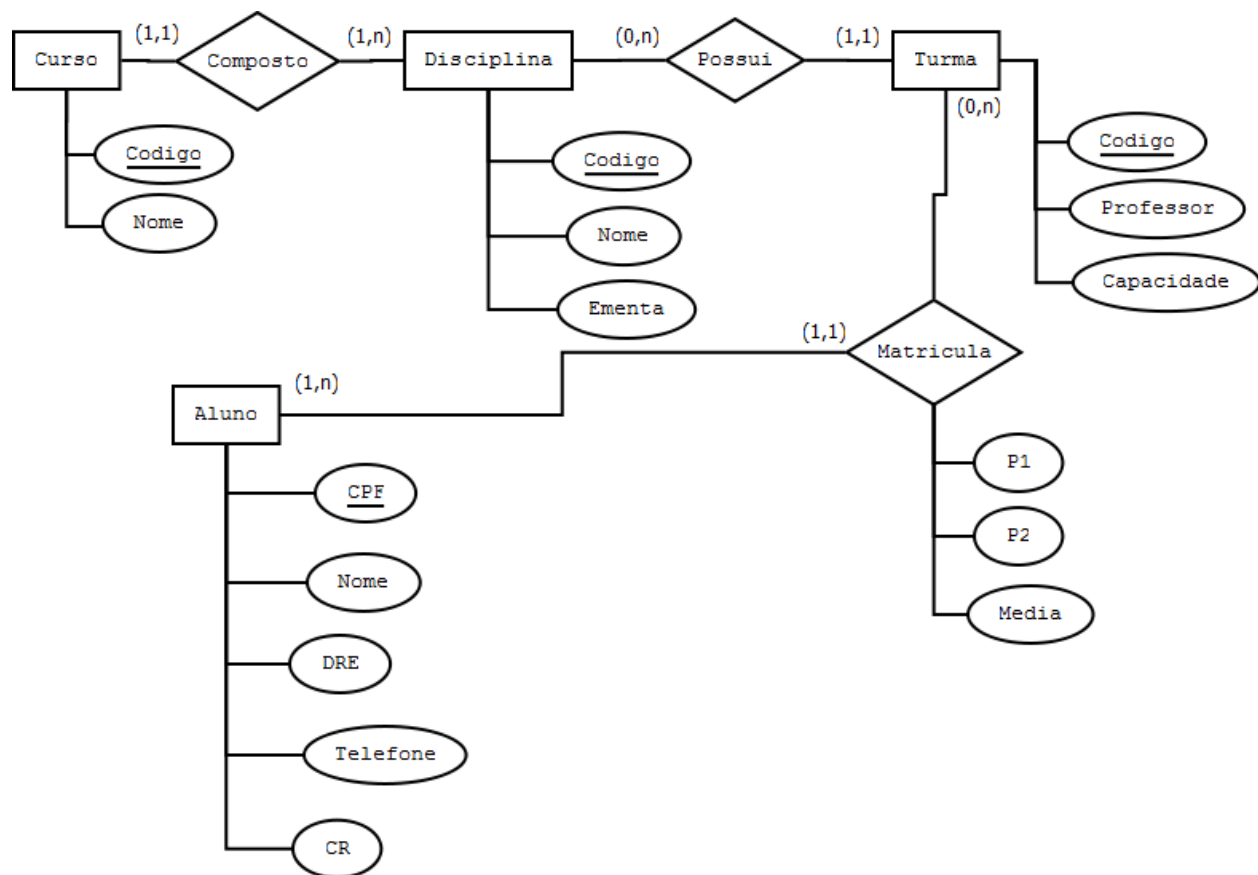
(d) Faça uma consulta SQL que retorna a descrição das peças que foram usadas no upgrade do computador de número de série M4N68T-M. [0.2 ponto]

```
SELECT p.descricao_peca
FROM Peca p, Peca_Upgrade_Revisao pur
WHERE p.cod_peca = pur.cod_peca
AND pr.no_serie = "M4N68T-M"
```

(e) Faça uma consulta SQL que retorna o nome, o cpf do cliente e a quantidade de computadores que fazem revisão ou upgrade na empresa. [0.2 ponto]

```
SELECT c.cpf, c.nome, COUNT(*)
FROM Cliente c, Computador co
WHERE c.cpf = co.cpf
GROUP BY c.cpf, c.nome
```

**Questão 6 (1 ponto).** Considere o diagrama ER mostrado abaixo, relativo a um sistema de controle de matrículas de alunos em um curso universitário. Construa um esquema relacional equivalente a este diagrama ER. O diagrama encontra-se na notação do DIA (ferramenta que usamos nas aulas para construir modelos ER).



**CURSO (ccod, nome);**

**DISCIPLINA (dcod,nome,ementa,ccod)**  
**(ccod) references CURSO**

**TURMA (tcod, professor, capacidade, dcod)**  
**(dcod) references DISCIPLINA**

**ALUNO (cpf,nome,dre,telefone,cr)**

**MATRICULA (cpf,tcod,p1,p2,media)**  
**(cpf) references ALUNO**  
**(tcod) references TURMA**

**Questão 7 (1 ponto).** A vídeo Center of Europe Ltda. é uma cadeia de locadoras de DVDs. Ela precisa manter dados sobre os DVDs que têm para locação, os filmes dos DVDs, e locadoras da rede e suas respectivas localidades. Cada DVD para locação tem um número de série único. Um projetista de banco de dados inexperiente propôs a seguinte tabela como solução para armazenar os dados da rede de locadoras. Suponha que cada filme tenha apenas um diretor (as chaves primárias estão sublinhadas):

Locadora (CodLocadora, Nome, CodLocal, NomeLocal (no\_serie, Filme, Diretor, Ano, Tipo))

O significado de cada coluna é o seguinte:

- CodLocadora: código da locadora
- Nome: nome da locadora
- CodLocal: código da localidade onde a locadora se localiza
- NomeLocal: nome da localidade onde a locadora se localiza
- no\_serie: Número de série do DVD
- Filme: nome do filme
- Diretor: nome do diretor
- Ano: ano de filmagem do filme
- Tipo: classificação do filme (ação, policial, terror, etc.)

As dependências funcionais (podendo incluir dependências transitivas) que existem nesta tabela são as seguintes:

CodLocadora → Nome

CodLocal → NomeLocal

CodLocadora → CodLocal

(CodLocadora, no\_serie) → Filme, Diretor, Ano, Tipo

Filme → Diretor

1. Assumindo que o profissional não conhece o conceito de normalização, explique para ele em que forma normal encontra-se a tabela.

**Não se encontra normalizada.**

2. Caso a tabela não se encontre na terceira forma normal, ensine ao profissional como a transformar para a terceira forma normal. Mostre cada forma normal intermediária, entre aquela em que a tabela se encontra e a terceira forma normal.

**Passagem para a Primeira Forma Normal**

**Locadora (CodLocadora, Nome, CodLocal, NomeLocal)**

**DVD (CodLocadora, no\_serie, Filme, Diretor, Ano, Tipo)**

**Passagem para a Segunda Forma Normal**

**Já está.**

**Passagem para a Terceira Forma Normal**

**Locadora (CodLocadora, Nome, CodLocal)**

**Localidade (CodLocal, NomeLocal)**

**DVD (no\_serie, Filme)**

**Filme (Filme, Diretor, Ano, Tipo)**