

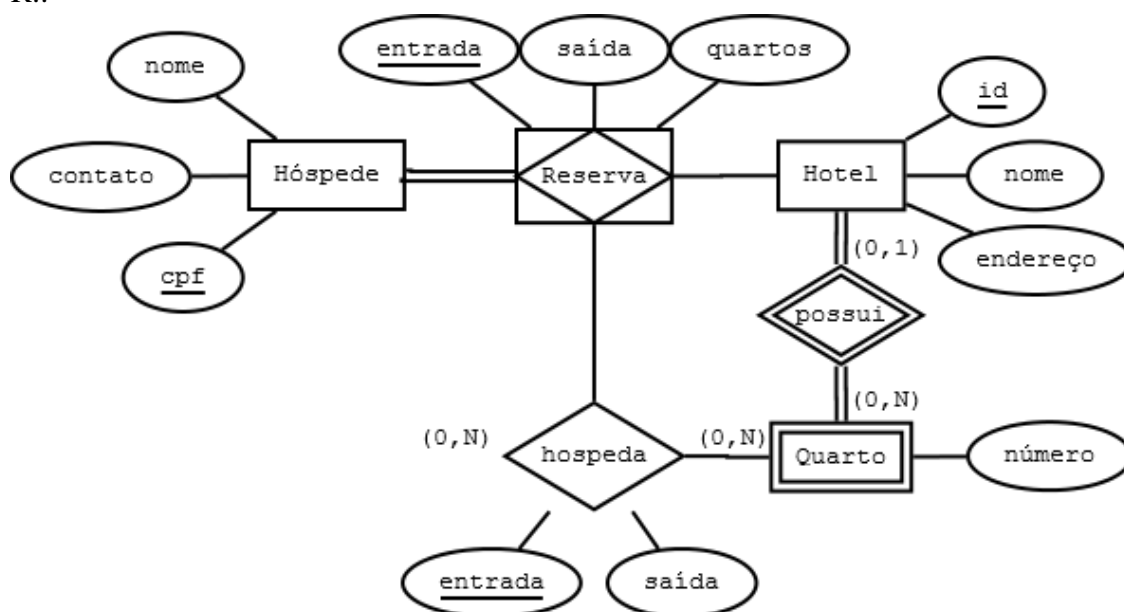
<b>Q1 (3,0)</b>	
<b>Q2 (3,0)</b>	
<b>Q3 (2,0)</b>	
<b>Q4 (1,0)</b>	
<b>Q5 (1,0)</b>	
<b>Nota:</b>	

Aluno: \_\_\_\_\_

Matrícula: \_\_\_\_\_ Turma: \_\_\_\_\_

- 1) [3,0 pontos] Uma rede de hotéis só permite hospedagens mediante reservas. Para fazer uma reserva os clientes precisam informar seus dados pessoais, hotel desejado, período de estadia e quantidade de quartos desejada. As reservas poderão se concretizar com uma hospedagem no hotel informado. As reservas e hospedagens somente poderão se concretizar caso exista disponibilidade de quartos. Faça os modelos de dados conceitual (1,0 pontos) e lógico (1,0 pontos) para uma aplicação de controle de reservas e hospedagens. Acrescente também as restrições de integridade referencial (1,0 pontos). Utilize somente atributos essenciais.

R.:



*Hospede* (cpf<sub>1</sub>, nome, contato)

*Hotel* (id<sub>1</sub>, nome, endereço)

*Reserva* (hospede<sub>1</sub>, hotel<sub>1</sub>, entrada<sub>1</sub>, saida, quartos)

*hospede referencia Hospede*

*hotel referencia Hotel*

*Quarto* (hotel<sub>1</sub>, numero<sub>1</sub>)

*hotel referencia Hotel*

*Hospeda* (hospede<sub>1</sub>, hotel<sub>1</sub>, entrada<sub>1</sub>, numero<sub>1</sub>, saida)

*hospede, hotel, entrada referenciam Reserva*

*hotel, numero referenciam Quarto*

- 2) [3,0 ponto] Considerando o esquema lógico do banco de dados apresentado a seguir, especifique uma consulta SQL para listar o CGC e o nome das revendedoras que venderam carros japoneses ou americanos, fabricados em 1995, por valor até 30% acima do preço de tabela.

*Automoveis (código, fabricante, modelo, ano, país, preço\_tabela)*

*Revendedoras (cgc, nome, proprietário, cidade, estado)*

*Negocios (cpf, cgc, codigo, ano, data, preço)*

R.:

**with**

```
    t1 as (select distinct r1.cgc
           from Negocios as r1
           inner join Automoveis as r2 on r1.codigo=r2.codigo
           where preco <= (preco_tabela * 1.3)
           and (fabricante='Japao' or fabricante='EUA')
           and r2.ano=1995 )
select r2.cgc, r2.nome from t1 natural join Revendedoras as
r2;
```

- 3) [2,0 pontos] Considerando o esquema lógico de banco de dados a seguir, escreva uma consulta SQL recursiva para listar a matrícula e o nome dos empregados subordinados direta ou indiretamente a outro empregado de matrícula M.

*Unidade(codigo,nome,super)*

*super referencia Unidade*

*Empregado(matricula,nome,unidade)*

*unidade referencia Unidade*

R.:

**with**

```
    recursive
    t1(unidade) as (select codigo
                    from unidade
                    where super=123
                    union
                    select r1.codigo
                    from unidade as r1
                    inner join t1 on r1.super=t1.unidade)
select r1.matricula, r1.nome from t1 natural join empregado
as r1
```

- 4) [1,0 ponto] Defina a forma normal de Boyce/Codd em termos das dependências funcionais.

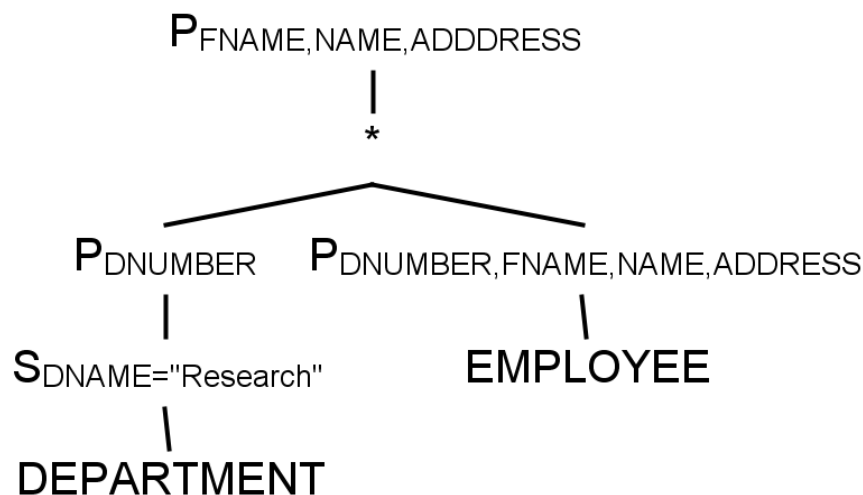
R.: **Uma tabela encontra-se na forma normal Boyce/Codd quando todo determinante for uma chave.**

5) [1,0 ponto] Considere a seguinte consulta SQL ao banco de dados Empregado X Projeto.

```
SELECT E.FNAME, E.NAME, E.ADDRESS
FROM EMPLOYEE E, DEPARTMENT D
WHERE D.NAME='Research' AND D.DNUMBER=E.DNUMBER;
```

Apresente a árvore algébrica otimizada, somente com as operações de junção ( $\bowtie$ , \*), seleção ( $\sigma$ ), projeção ( $\pi$ ) e união ( $\cup$ ), correspondente ao comando SQL acima. Considere os seguintes critérios para otimização: (a) Realize o mais cedo possível as operações de seleção; (b) Projete cada relação e o resultado de cada operação de forma a manter apenas os atributos indispensáveis às operações seguintes; e (c) Realize o mais cedo possível as operações mais restritivas. Considere que a ordem de execução das operações é estabelecida pelo percurso na árvore em profundidade à esquerda. Não é necessário apresentar o processo de otimização passo a passo, basta apresentar a árvore algébrica final otimizada.

R.:



EMPLOYEE	FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
	John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
	Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
	Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
	Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
	Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
	Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
	Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
	James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	null	1

DEPARTMENT	DNAME	DNUMBER	MGRSSN	MGRSTARTDATE
Research		5	333445555	1988-05-22
Administration		4	987654321	1995-01-01
Headquarters		1	888665555	1981-06-19