



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**

**Disciplina Banco de Dados**

**AP3 2º semestre de 2010.**

**Nome –**

**Assinatura –**

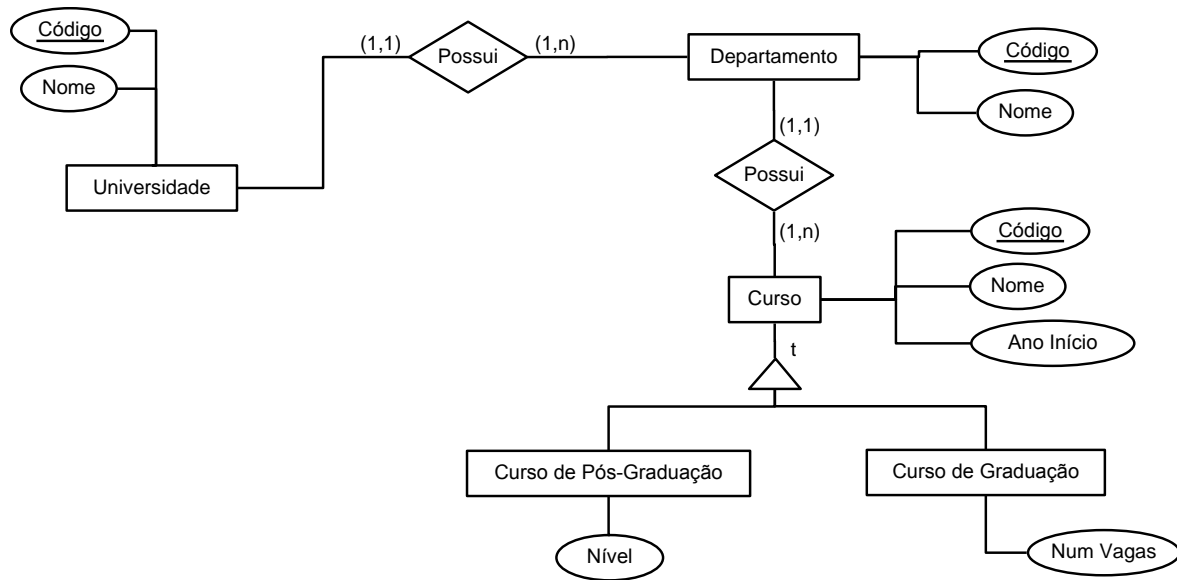
---

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
  2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
  3. Você pode usar lápis para responder as questões.
  4. Ao final da prova devolva as folhas de questões e as de respostas.
  5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
- 

**Questão 1** [2 pontos]

Considere o diagrama ER mostrado abaixo, que modela um sistema que armazena informações sobre cursos de universidades. O diagrama encontra-se na notação do DIA (ferramenta que usamos em sala de aula para construir modelos ER). Construa um esquema relacional equivalente a este diagrama ER. A base de dados deve refletir exatamente o especificado no modelo conceitual. O esquema da base de dados relacional deve conter os nomes das tabelas, os nomes dos atributos, atributos que formam a chave primária e as chaves estrangeiras. Utilizar tabela única para mapear a generalização/especialização.



Resposta (as chaves primárias estão sublinhadas):

Universidade (Código, Nome)

Departamento (Código, Nome, CodUniversidade)

CodUniversidade referencia Universidade

Curso (Código, Nome, AnoInício, Nível, NumVagas, Tipo, CodDepto)

CodDepto referencia Departamento

## Questão 2 [3,0 pontos]

Considere a base de dados relacional com dados sobre pacientes e internações, cujo esquema encontra-se abaixo (as chaves primárias estão sublinhadas):

Paciente(CodPac, NomePac, DataNascPac)

/\* tabela de pacientes \*/

Internação(CodPac, NumeroIntern, DataBaixa, DataAlta, NumLeito)

CodPac referencia Paciente

NumLeito referencia Leito

/\* tabela com as internações já ocorridas no hospital \*/

Leito (NumLeito, NumQuarto, TipoLeito)

/\* tabela de leitos com respectivos quartos e tipos \*/

- (a) Escreva o comando SQL para criar a tabela *Leito*, incluindo as restrições de integridade que se aplicam. [0,5 ponto]

```
CREATE TABLE Leito (  
    NumLeito integer NOT NULL,  
    NumQuarto integer,  
    TipoLeito char,  
    Primary key (NoLeito)  
)
```

- (b) Escreva o comando SQL para adicionar a coluna Telefone na tabela *Paciente*. [0,5 ponto]

```
ALTER TABLE Paciente  
ADD Telefone VARCHAR(10)
```

- (c) Escreva um comando SQL para alterar a informação sobre Internação da paciente “Mariana Silva” cuja data de internação é “06/06/2010”. A alteração deve modificar a data de alta para “06/07/2010”. [1 ponto]

```
UPDATE Internação  
SET DataAlta="06/07/2010"  
WHERE DataBaixa = "06/06/2010" AND  
CodPac IN (SELECT CodPac  
            FROM Paciente  
            WHERE NomePac="Marina Silva")
```

- (d) Escreva um comando SQL para excluir todos os registros de internações que ocorreram em leito do tipo “A”. [1 ponto]

```
DELETE FROM Internação  
WHERE NumLeito IN (  
    SELECT NumLeito FROM Leito  
    WHERE TipoLeito="A"  
)
```

### Questão 3 [4 pontos]

Considere o esquema relacional da questão 2.

Apresente as expressões algébricas para as seguintes consultas:

- (a) Obter o código, nome do paciente e o número do leito em que o paciente está ou já esteve internado [0,5 ponto]

$\pi_{CodPac, NomePac, NumLeito} (Paciente \bowtie Internação)$   
 $CodPac = CodPac$

(b) Obter os nomes dos pacientes que ocupam/ocuparam o leito número 25 [0,5 ponto]

$\pi_{NomePac} (Paciente \bowtie_{\substack{CodPac = CodPac}} (\sigma_{NumLeito = 25}) Internação)$

(c) Obter os nomes dos pacientes que não estão internados. [0,5 ponto]

$\rho(R1, \pi_{NomePac}(Paciente))$   
 $\rho(R2, \pi_{NomePac}(Paciente \bowtie_{\substack{CodPac = CodPac}} Internação))$

$R1 - R2$

Apresente agora o código SQL para as seguintes consultas:

(d) Obter os números dos leitos que estão atualmente ocupados por alguma internação. Assuma que, quando um leito está ocupado, o atributo DataAlta é NULL. [0,5 ponto]

```
SELECT NumLeito
FROM      Internação
WHERE     DataAlta IS NULL
```

(e) Obter os nomes dos pacientes nascidos antes de 01/01/1990 e o número de vezes que estes pacientes já foram internados. [1,0 ponto]

```
SELECT CodPac, NomePac, COUNT (*)
FROM      Paciente p, Internação i
WHERE     p.CodPac = i.CodPac
AND       p.DataNascPac < 01/01/1990
GROUP BY p.CodPac, p.NomePac
```

(f) Obter os nomes dos pacientes que foram internados depois de 01/01/2010. [1,0 ponto]

```
SELECT NomePac
FROM      Paciente p, Internacao i
WHERE     p.CodPac = i.CodPac
AND       i.dataBaixa > 01/01/2010
```

**Questão 4** [1 ponto]

Dê um exemplo de uma tabela que não se encontra na segunda forma normal. Explique por que a tabela não está na segunda forma normal, e apresente todas as dependências funcionais que se aplicam.

Para esta questão, será aceito qualquer exemplo que contenha uma tabela que apresente dependência funcional parcial.

Emprestimo (CodUsuario, CodObra, NomeUsuario, TituloObra)

A tabela Emprestimo não se encontra na segunda forma normal porque possui as seguintes dependências parciais:

CodUsuario → NomeUsuario

CodObra → TituloObra