



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina Banco de Dados

AP3 1º semestre de 2008.

Nome –

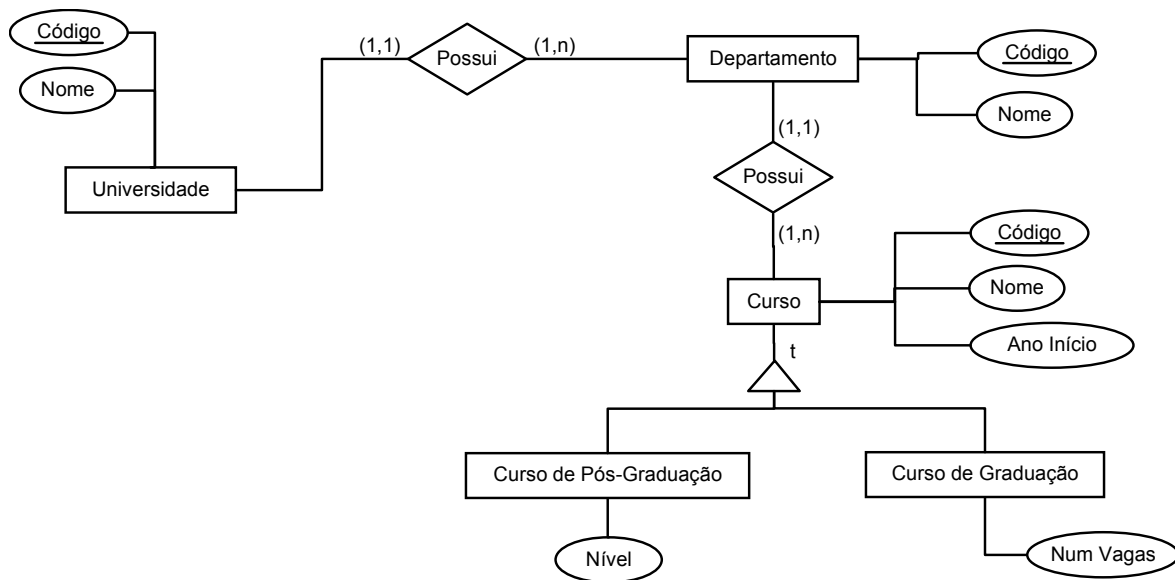
Assinatura –

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 com todos pacientes internados na atualidade ou no passado no hospital, inclusive os nele nascidos */

Internação(CodPac, NumeroIntern, DataHoraBaixa, DataHoraAlta)

CodPac referencia Paciente

/* tabela com as internações já ocorridas no hospital - para partos, há dois registros, um para a mãe, outro para o bebê */

Leito (NoLeito, CodigoQuarto, TipoLeito)

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

OcupaLeito(CodPac, NumeroIntern, NoLeito, DataHoraInic, DataHoraFim)

(CodPac, NumeroIntern) referencia Internação

NoLeito referencia Leito

/* tabela de ocupação de leitos indicando que paciente internado ocupou que leito */

Parto(CodPacBebe,NumeroInternBebe,HoraNasc,TipoParto,CodPacMae,NumeroInternMae)

(CodPacBebe,NumeroInternBebe) referencia Internação

(CodPacMae,NumeroIntern) referencia Internação

/* tabela de partos - CodPacBebe é o código de paciente recém nascido (bebê) -

CodPacMae é o código de paciente da mãe. Para cada um deles há uma internação. */

- (a) Escreva o comando SQL para criar a tabela *Internação*, incluindo as restrições de integridade que se aplicam. Assuma que um paciente não pode ser excluído se houver uma internação ligada a ele. Se, por algum motivo, não for possível definir alguma restrição de integridade, justifique. [0,5 ponto]

```
CREATE TABLE Internacao (  
    CodPac integer NOT NULL,  
    NumeroIntern integer NOT NULL,  
    DataHoraBaixa time,  
    DataHoraAlta time  
    Primary key (CodPac) references Paciente ON DELETE RESTRICT  
)
```

- (b) Escreva o comando SQL para excluir a tabela *OcupaLeito*. [0,5 ponto]

```
DROP TABLE OcupaLeito
```

- (c) Escreva um comando SQL para alterar a internação do paciente “Marina Silva” realizada em “01/06/2007 15:00”. A alteração deve modificar a data/hora de alta do paciente para “06/06/2007 17:00”. [1 ponto]

```
UPDATE Internacao  
SET DataHoraAlta="06/06/2007 17:00"  
WHERE DataHoraBaixa = "01/06/2007 15:00" AND  
CodPac IN (SELECT CodPac  
    FROM Paciente  
    WHERE NomePac="Marina Silva")
```

- (d) Escreva um comando SQL para excluir informações sobre o parto em que nasceu “João Victor”. Suponha que não existe DELETE CASCADE, e portanto, apresente o código SQL necessário para excluir todas as informações sobre este parto, incluindo ocupação de leito (somente do bebê), e internação (somente do bebê). Assuma que só existe uma única internação para este bebê. Os dados dos pacientes envolvidos (mãe e bebê) devem ser preservados. Apresente as instruções SQL na ordem em que devem ser executadas para não gerar erro de integridade referencial. [1 ponto]

```
DELETE FROM Parto  
WHERE CodPacBebe IN (  
    SELECT CodPac FROM Paciente
```

```

WHERE NomePac = "João Vitor"
);

DELETE FROM OcupaLeito
WHERE CodPacBebe IN (
    SELECT CodPac FROM Paciente
    WHERE NomePac = "João Vitor"
);

DELETE FROM Internação
WHERE CodPacBebe IN (
    SELECT CodPac FROM Paciente
    WHERE NomePac = "João Vitor"
);

```

Questão 3 [4 pontos]

Considere o esquema relacional da questão 2.
 Apresente as expressões algébricas para as seguintes consultas:

(a) Obter o código, nome e data de nascimento dos pacientes internados [0,5 ponto]

$\pi_{\text{CodPac}, \text{NomePac}, \text{DataNascPac}} (\text{Paciente} \bowtie_{\text{CodPac} = \text{CodPac}} \text{Internação})$

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

$\pi_{\text{NomePac}} (\text{Paciente} \bowtie_{\text{CodPac} = \text{CodPac}} (\sigma_{\text{NumLeito} = 25} \text{OcupaLeito}))$

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

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

R1 - R2

Apresente agora o código SQL para as seguintes consultas:

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

```

SELECT NoLeito
FROM      OcupaLeito
WHERE     DataHoraFim IS NULL

```

(e) Obter os nomes dos pacientes nascidos antes de 01/01/1985, que tiveram mais que três internações. Junto com o nome do paciente, mostrar o número de internações. Obter o resultado em ordem decrescente de número de internações. [0,5 ponto]

```
SELECT NomePac, COUNT (*)
FROM      Paciente P , Internação I
WHERE      P.CodPac = I.CodPac
AND        DataNascPac < 01/01/1985
GROUP BY P.CodPac
HAVING COUNT (*) > 3
ORDER BY DESC COUNT(*)
```

(f) Obter os nomes dos pacientes que ocupam leitos do tipo 1. [0,5 ponto]

```
SELECT NomePac
FROM      Paciente P , OcupaLeito I, Leito L
WHERE      P.CodPac = I.CodPac
AND        I.NoLeito = L.NoLeito
AND        TipoLeito = 1
```

(g) Escreva um comando SQL para criar uma visão chamada MAES_FILHOS que contenha o nome da mãe e o nome de todos os seus filhos. [1 ponto]

```
CREATE VIEW MAES_FILHOS (NomeMae, NomeFilho) AS
SELECT p1.NomePac, p2.NomePac
FROM Paciente p1, Paciente p2, Parto p
WHERE p1.CodPac = p.CodPacMae
AND p2.CodPac = p.CodPacBebe
```

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.

Empréstimo (CodUsuario, CodObra, NomeUsuario, TituloObra)

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

CodUsuario → NomeUsuario

CodObra → TituloObra