



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**

**Disciplina Banco de Dados**

**AP1 2º semestre de 2018**

**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 [4,0 pontos]**

Deseja-se projetar uma base de dados que dará suporte a um sistema para controlar as horas trabalhadas pelos profissionais de uma empresa de consultoria. O sistema destina-se a coletar dados para cobrança dos clientes da empresa. Utilizando um diagrama entidade-relacionamento, deve ser modelada esta base de dados. A base de dados não deve conter redundância de dados. O modelo ER deve ser representado com a notação vista em aula ou com outra notação de poder de expressão equivalente. O modelo deve apresentar, ao menos, entidades, relacionamentos, atributos, especializações, identificadores e restrições de cardinalidade. Não usar atributos multivalorados. O modelo deve ser feito no nível conceitual, sem incluir chaves estrangeiras.

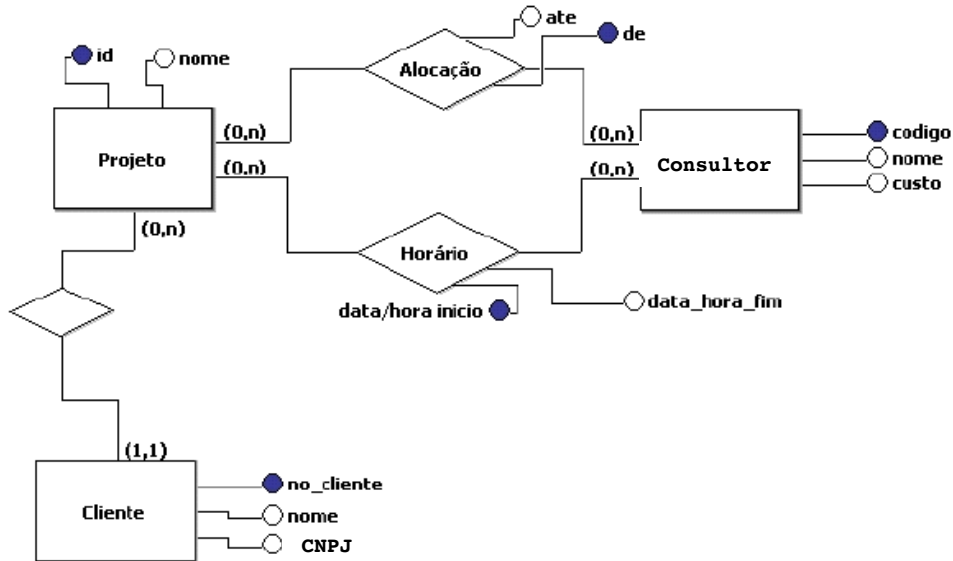
Todas atividades da empresa acontecem por meio de projetos. Para cada projeto, o banco de dados deve armazenar um identificador, o nome do projeto e o seu cliente. Um cliente pode ter vários projetos. Além dos projetos do cliente, o banco de dados deve armazenar o número (único) do cliente, seu CGC e seu nome.

Em cada projeto são alocados vários consultores. Um consultor é alocado a um projeto por um tempo determinado (de-até). Cada consultor tem um código identificador, um nome e um custo por hora trabalhada.

Para cada vez que um consultor trabalha em um projeto, mesmo que por alguns minutos, o banco de dados deve armazenar, além do projeto e do consultor, a data/hora em que o

consultor começou a trabalhar e a data/hora em que ele encerrou o trabalho. É preciso manter o histórico dos projetos em que um determinado empregado trabalhou.

**Resposta:**



### **Questão 2 (2 pontos)**

Explique as seguintes vantagens decorrentes de se utilizar um Sistema de Gerência de Bases de Dados (SGBD) em relação a se usar diretamente operações sobre diretórios e arquivos do sistema operacional.

- Redundância controlada dos dados
- Independência dos dados

**Resposta:**

- Num cenário de programas de aplicações que utilizam sistemas de arquivos para gerenciar os dados, pode ocorrer redundância entre os dados armazenados. Normalmente os dados estão dispersos em diversos arquivos, com redundância parcial ou total. Por exemplo, uma empresa pode ter dois arquivos de cadastro de clientes: um que é acessado pelo setor de vendas, e outro que é acessado pelo setor financeiro. Essa redundância é dita não controlada, uma vez que ela ocorre de modo independente – a responsabilidade pela manutenção da consistência dos dados é do usuário do sistema. Já num SGBD, ambas as aplicações usariam a mesma tabela, já que ele provê compartilhamento de dados e processamento de transações multi-usuário – isso evita a redundância não controlada. Ainda assim, existe uma redundância mínima no armazenamento dos dados usados por programas de aplicações. Em geral, os atributos chave primária e estrangeira aparecem tanto na tabela que

referencia quanto na tabela referenciada. Entretanto, essa redundância é controlada de forma automática pelo SGBD.

- b) Programas de aplicação independem dos detalhes de representação e armazenamento dos dados. A independência lógica de dados significa que os usuários ficam imunes às mudanças na estrutura lógica dos dados, enquanto que a independência física de dados isola os usuários de mudanças no armazenamento físico dos dados. Os esquemas físico, lógico e externo com seus mapeamentos provêm independência das decisões quanto ao armazenamento físico e projeto lógico respectivamente. Esquemas externos permitem que o acesso aos dados seja particularizado (e autorizado) ao contexto individual dos usuários ou grupos de usuários. Esquemas lógicos descrevem todos os dados que estão armazenados numa determinada base de dados. Enquanto existem inúmeras visões para uma mesma base de dados, existe apenas um único esquema lógico para todos os usuários de uma base de dados. Esquemas internos (físicos) definem como são armazenadas (no disco ou em outro meio físico) as relações descritas no esquema lógico. Como um exemplo, suponha que, por necessidade de uma aplicação específica, precisa-se adicionar um atributo telefone a uma relação pré-existente ALUNO (aid, anome, cra). Os programas de aplicações que operam sobre a relação ALUNO ficam imunes a essa mudança, uma vez que não dependem da estrutura física dos dados.

### **Questão 3 [3 pontos]**

Considere a seguinte base de dados, usada por uma sapataria. O cliente entrega um sapato para conserto e o retira algum tempo depois de o serviço ser prestado. As chaves primárias estão sublinhadas.

```
/* tabela de clientes cadastrados na sapataria */
CLIENTE (cpf, nomeCli, celular)

/* tabela com dados dos sapatos dos clientes da sapataria – ao chegar na sapataria, cada sapato
recebe um código de barras único para facilitar a sua identificação*/
SAPATO (codBarra, numero, modelo, cor, cpf);
(cpf) referencia CLIENTE

/* tabela com os reparos feitos */
REPARO (codBarra, dataExecutada, dataUltimoConserto)
(codBarra) referencia SAPATO

/* tabela com os materiais usados em cada reparo */
MATERIAL_REPARO (codBarra, dataExecutada, codMat, quantidade)
(codBarra, dataExecutada) referencia REPARO
(codMat) referencia MATERIAL

/* tabela com as descrições dos materiais */
MATERIAL (codMat, descricao)
```

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

(a) Faça uma consulta que retorna os modelos dos sapatos do cliente chamado “João Moreira” [1 ponto].

$\pi_{\text{modelo}} (\text{Sapato} * (\sigma_{\text{nomeCli} = \text{“João Moreira”}} \text{Cliente}))$

(b) Faça uma consulta que retorna a descrição do material utilizado no reparo realizado no dia 01/03/2018 do sapato cujo código de barras é 000111 [1 ponto].

$\pi_{\text{descricao}} (\text{Material} * (\text{MaterialReparo} * (\sigma_{\text{dataExecutada} = \text{“01/03/2018”}} \text{Reparo} * (\sigma_{\text{codBarra} = \text{“000111”}} \text{Sapato})))$

(c) Faça uma consulta que mostre a descrição dos materiais que já foram utilizados em algum reparo [1 ponto].

$\pi_{\text{descricao}} (\text{Material} * (\text{MaterialReparo}))$

Note que aqui a tabela Reparo não é necessária.

#### **Questão 4 [1 ponto]**

Utilizando o esquema da questão 3, analise as consultas a seguir e diga qual o esquema da tabela retornada por cada uma das consultas. O esquema deve ser informado utilizando a seguinte sintaxe:

Tab (TabOrigem<sub>1</sub>.Atrib<sub>1</sub>, TabOrigem<sub>1</sub>.Atrib<sub>2</sub>, ..., TabOrigem<sub>N</sub>.Atrib<sub>1</sub>, TabOrigem<sub>N</sub>.Atrib<sub>2</sub>, ...)

Nesta notação, tabOrigem<sub>i</sub> é o nome da tabela de onde veio o atributo Atrib<sub>j</sub> originalmente.

(a)  $\sigma_{\text{nomeCli} = \text{“Vanessa”}} (\text{Cliente})$

Tab (Cliente.cpf, Cliente.nomeCli, Cliente.celular)

(b)  $(\pi_{\text{nomeCli, cpf}} (\text{Cliente})) * \text{Sapato} * (\sigma_{\text{dataUltimoConserto} < \text{“10/03/2014”}} \text{Reparo})$

Tab (Cliente.nomeCli, Cliente.cpf, Sapato.codBarra, Sapato.numero, Sapato.modelo, Sapato.cor, Reparo.dataExecutada, Reparo.dataUltimoConserto)