

Lista de Exercícios

MC536 - Bancos de Dados: Teoria e Prática
 Instituto de Computação
 Universidade Estadual de Campinas

Transações, Concorrência, Consultas e Stored
 Procedures (questões estilo prova)
 2013
 André Santanchè

Questão 1

Considere a seguinte tabela que registra saldos de conta corrente, em que é registrado o id da conta e seu saldo. A tabela só tem três registros apresentados abaixo:

| ContaCorrente | |
|---------------|-------|
| contaid | saldo |
| X | 500 |
| Y | 200 |
| Z | 350 |

```
CREATE TABLE ContaCorrente (
  contaid VARCHAR(5) NOT NULL ,
  saldo FLOAT,
  PRIMARY KEY (contaid) );

INSERT INTO ContaCorrente VALUES ("X", 500);
INSERT INTO ContaCorrente VALUES ("Y", 200);
INSERT INTO ContaCorrente VALUES ("Z", 350);
```

Considere as duas transações sendo executadas em paralelo sobre estas tabelas em que podem ser intercaladas operações de leitura e gravação.

| T1: Transferência da conta Y para a conta X | T2: Somatório de todos os saldos de conta |
|---|--|
| <pre>UPDATE ContaCorrente SET saldo = saldo - 100 WHERE contaid = "Y"; UPDATE ContaCorrente SET saldo = saldo + 100 WHERE contaid = "X";</pre> | <pre>CREATE VIEW SaldoTotal AS SELECT SUM(saldo) FROM ContaCorrente;</pre> |

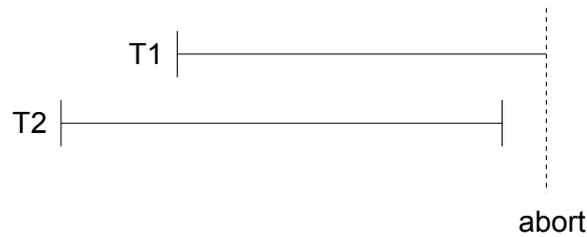
Para cada uma das questões a seguir que solicitar um plano de execução, utilize o identificador da transação (T1: ou T2:) antes de cada instrução no plano. As instruções que podem ser usadas no plano (quando necessário) são: read (leitura), write (gravação), lock/unlock (bloqueio binário), rlock/wlock/unlock (bloqueio compartilhado exclusivo), start (início transação), commit (transação recebe commit), abort (transação abortada). Cada instrução que envolver uma tupla da tabela deve indicar a tupla pela sua chave primária. Exemplos de operações em um plano:

| |
|-----------------------------|
| T1: read(X) T2: write(Y) |
|-----------------------------|

Responda as seguintes questões a seguir (na prova o aluno escolheu duas entre três para responder):

- a) Monte um plano de execução que apresente problema de isolamento. Indique de forma sintética qual o problema. Indique que característica deve ter um plano para evitar este problema e apresente o plano corrigido.
- b) Apresente um plano com problemas de deadlock (explícite os bloqueios e desbloqueios no plano), indique onde será o deadlock e indique que tipo de plano 2PL pode ser adotado para evitá-lo e porquê.

c) Considere o diagrama a seguir ilustrando uma possível execução concorrente de T1 e T2:



Há algum tipo de plano de execução que possa ser apresentado aproximadamente conforme este diagrama e que tenha problemas de atomicidade se acontecer o abort indicado? Que regra um plano deve seguir para que nunca tenha este problema.

Questão 2

Uma consulta antes de ser executada pelo banco de dados passa por vários passos e um deles é o Otimizador de consulta, nessa fase o otimizador converte a consulta SQL em uma equivalente em álgebra relacional e cria uma árvore de operações a fim de obter a melhor estratégia de execução.

Considere as seguintes relações:

Aluno (alunoid, alunonome)

Curso (cursoid, cursonome)

Matricula (alunoid, cursoid, ano)

Construa a árvore de operações para a seguinte consulta com as respectivas otimizações realizadas por um otimizador de consulta:

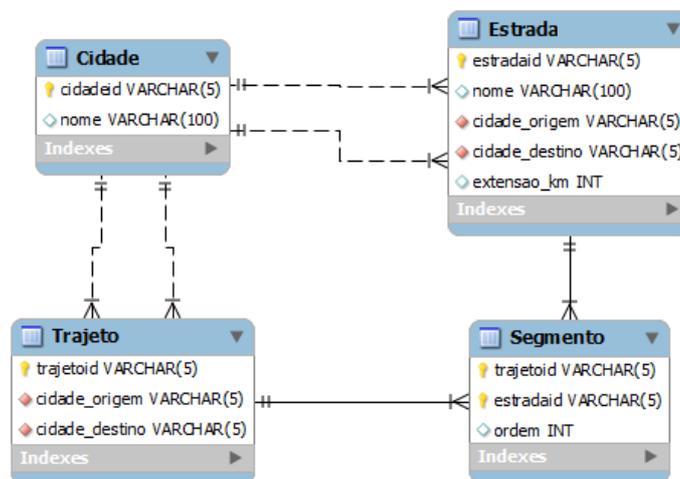
```
SELECT A.name
FROM Aluno A, Matricula M, Curso C
WHERE A.alunoid = M.alunoid and C.cursoid = M.cursoid and
      C.nome = "Banco de dados" and M.ano = 2011
```

Questão 3

Considere os comandos SQL abaixo para criar tabelas que controlam cidades, estradas e trajetos entre cidades. A tabela de Cidades mantém um cadastro de cidades; a tabela de Estradas registra estradas que ligam uma cidade (cidade_origem) a outra (cidade_destino), bem como sua quilometragem. Cada registro da tabela Trajeto especifica um trajeto, que consiste em uma sequência ordenada de estradas que ligam duas cidades (cidade_origem e cidade_destino), por exemplo, um trajeto entre Salvador e Curitiba, pode envolver uma sequência de estradas: Salvador-Belo Horizonte, Belo Horizonte-São Paulo e São Paulo-Curitiba. A tabela Segmento associa estradas a trajetos. O campo ordem é um campo numérico sequencial (iniciado de 1 para cada trajeto) usado para ordenar os segmentos (estradas) dentro de um trajeto.

| | |
|--|--|
| <pre>CREATE TABLE Cidade (cidadeid VARCHAR(5) NOT NULL , nome VARCHAR(100) , PRIMARY KEY (cidadeid));</pre> | <pre>CREATE TABLE Trajeto (trajetoid VARCHAR(5) NOT NULL , cidade_origem VARCHAR(5) NOT NULL , cidade_destino VARCHAR(5) NOT NULL , PRIMARY KEY (trajetoid) , FOREIGN KEY (cidade_origem) REFERENCES Cidade (cidadeid) ON DELETE NO ACTION ON UPDATE NO ACTION, FOREIGN KEY (cidade_destino) REFERENCES Cidade (cidadeid) ON DELETE NO ACTION ON UPDATE NO ACTION);</pre> |
| <pre>CREATE TABLE Estrada (estradaid VARCHAR(5) NOT NULL , nome VARCHAR(100) , cidade_origem VARCHAR(5) NOT NULL , cidade_destino VARCHAR(5) NOT NULL , extensao_km INT , PRIMARY KEY (estradaid) , FOREIGN KEY (cidade_origem) REFERENCES Cidade (cidadeid) ON DELETE NO ACTION ON UPDATE NO ACTION, FOREIGN KEY (cidade_destino) REFERENCES Cidade (cidadeid) ON DELETE NO ACTION ON UPDATE NO ACTION);</pre> | <pre>CREATE TABLE Segmento (trajetoid VARCHAR(5) NOT NULL , estradaid VARCHAR(5) NOT NULL , ordem INT , PRIMARY KEY (trajetoid, estradaid) , FOREIGN KEY (estradaid) REFERENCES Estrada (estradaid) ON DELETE NO ACTION ON UPDATE NO ACTION, FOREIGN KEY (trajetoid) REFERENCES Trajeto (trajetoid) ON DELETE NO ACTION ON UPDATE NO ACTION);</pre> |

O diagrama abaixo representa graficamente o modelo relacional criado pelos comandos SQL (Obs.: este não é um diagrama ER).



A partir do esquema apresentado, escreva as seguintes consultas SQL:

- Qual o nome das cidades que não aparecem na origem de nenhum segmento.
- Escreva uma Stored Procedure que receba como parâmetros o id de um trajeto e o id de uma estrada e insira um registro na tabela Segmento incluindo a estrada informada como último segmento do trajeto. Considere que os ids pertencem a um trajeto e uma estrada que já estão cadastrados. O número de ordem do segmento não é passado como parâmetro e deve ser calculado automaticamente pela Stored Procedure.

Não podem ser usadas VIEWS na resolução destas questões.