

BCC441 Lista de Exercícios I - Processamento de Transações

Alunos: Mateus Lana e Thiago Santana

Matrículas: 15.1.4340 e 15.1.4313

Perguntas de Revisão

21.2.

As falhas podem ser classificadas como de transação, sistema e mídia. Os possíveis motivos para uma transação falhar podem ser:

- **Uma falha do computador (falha do sistema):** Um erro de hardware, software ou rede no sistema de computação durante a execução da transação
- **Um erro de transação ou do sistema:** Alguma operação na transação, como um estouro de inteiro ou divisão por zero, bem como valores errôneos de parâmetros, um erro lógico de programação ou até mesmo uma interrupção realizada pelo usuário.
- **Erros locais ou condições de exceção detectadas pela transação:** Podem ocorrer certas condições que necessitam de cancelamento da transação. Por exemplo, os dados da transação podem não ser encontrados.
- **Imposição de controle de concorrência:** O método de controle de concorrência pode decidir abortar uma transação porque ela viola a serialização ou pode abortar uma ou mais transações para resolver um estado de deadlock entre várias transações.
- **Falha de disco:** Alguns blocos de disco podem perder seus dados devido a um defeito de leitura, gravação ou por causa de uma falha da cabeça de leitura/gravação.
- **Problemas físicos e catástrofes:** Isso se refere a uma lista sem fim de problemas que incluem falha de energia ou de ar-condicionado, incêndio, roubo, sabotagem, regravação de discos ou fitas por engano e montagem da fita errada pelo operador

Uma falha catastrófica é aquela que ocasiona a perda do disco e sendo assim não pode ser corrigida pelo SGDB.

21.3.

A execução de um comando `read_item (X)` inclui as seguintes etapas:

1. Ache o endereço do bloco de disco que contém o item X.
2. Copie este bloco de disco para um buffer na memória principal (caso ele ainda não esteja em algum buffer na memória principal).
3. Copie o item X do buffer para a variável de programa chamada X.

A execução do `write_item (X)`, inclui os dois primeiros passos do comando `read_item`, e acrescenta os seguintes passos:

1. Copie o item X da variável de programa chamada X para o local correto no buffer.
2. Armazena o bloco atualizado do buffer de volta no disco (imediatamente ou em algum momento posterior).

21.6.

Atomicidade: Uma transação é uma unidade de processamento atômica; ela deve ser realizada em sua totalidade ou não ser realizada de forma alguma.

Preservação da consistência: Uma transação deve preservar a consistência, significando que, se ela for completamente executada do início ao fim sem interferência de outras transações, deve levar o banco de dados de um estado consistente para o outro.

Isolamento: A execução de uma transação não deve ser interferida por quaisquer outras transações que acontecem simultaneamente.

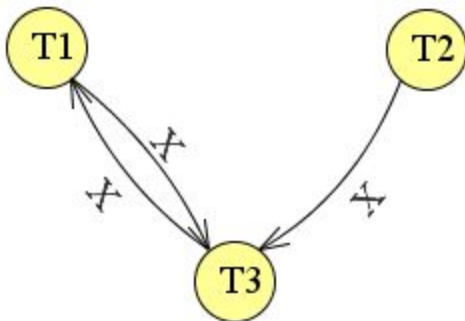
Durabilidade ou permanência: As mudanças aplicadas ao banco de dados pela transação confirmada precisam persistir no banco de dados. Essas mudanças não devem ser perdidas por causa de alguma falha.

Exercícios

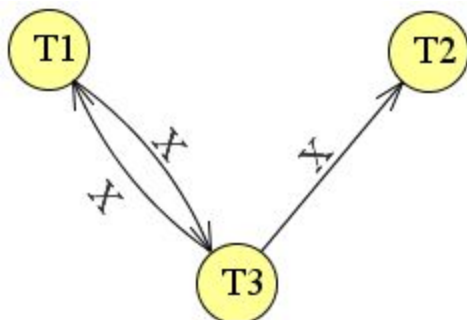
21.20. Pois uma transação começa quando for executada a 1ª operação SQL executável e termina com um COMMIT ou ROLLBACK. Os comandos de fim de transação devem ser explícitos para que as alterações realizadas por uma transação sejam efetivadas no banco de dados. Quando uma transação termina, o próximo comando SQL inicia automaticamente a próxima transação.

21.22.

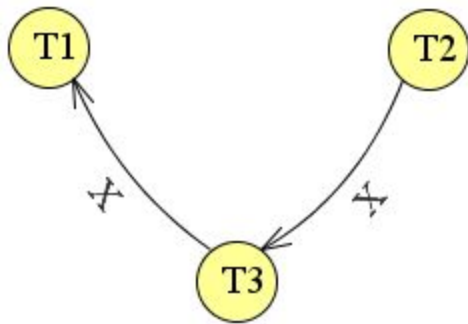
a. Não é serializável



b. Não é serializável



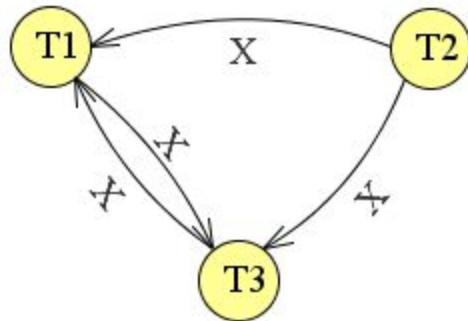
c. Serializável



Schedule serial equivalente:

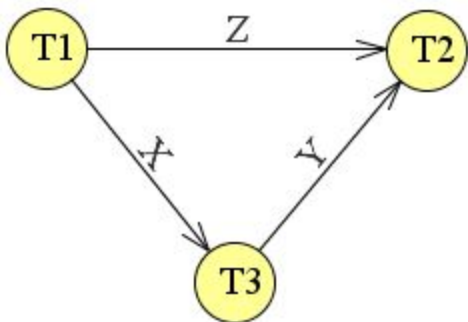
$r_3(X); w_3(X); r_2(X); r_1(X); w_1(X)$

d. Não é serializável



21.23.

S_1 : Não serializável



S_2 : Não serializável

