

AS10: Transação e Concorrência

- Entrega 14 jun em 8:40
- Pontos 3
- Perguntas 2
- Disponível 14 jun em 7:40 - 14 jun em 8:40 1 hora
- Limite de tempo 60 Minutos

Instruções

Teste

Este é o teste **AS10: Transação e Concorrência**, uma atividade prática avaliativa para testar o conhecimento do aluno em **conceitos e técnicas relacionados ao processamento de transações e controle de concorrência em bancos de dados relacionais**.

Instruções

De forma **individual** e **sem consulta**, o aluno deverá responder as questões apresentadas no teste **observando o limite de tempo** para sua conclusão. O aluno deverá responder **uma pergunta por vez** e **não terá a opção de voltar** para rever sua resposta ou responder questões não respondidas.

Este teste foi travado 14 jun em 8:40.

Pontuação deste teste: 3 de 3

Enviado 14 jun em 8:12

Esta tentativa levou 31 minutos.



Pergunta 1

1,5 / 1,5 pts

Abaixo apresentam-se três transações e um possível escalonamento envolvendo essas transações.

$$T_1 = r(x), r(y), w(x), r(z)$$

$$T_2 = r(z), r(x), r(y), w(z)$$

$$T_3 = r(y), r(z), w(y), r(x)$$

$$S_a = r_3(y), r_2(z), r_1(x), r_2(x), r_3(z), r_2(y), w_3(y), w_2(z), w_1(x), r_3(x), r_1(z)$$

Com base nas transações e nos escalonamento apresentados pode-se afirma que:

Correto!

☒ A transação T_2 é uma transação de leitura-escrita

☐ Não existem conflitos no escalonamento S_a

☐ O escalonamento S_a é completo

Correto!

☒ O escalonamento S_a não respeita as propriedades ACID

Correto!

☒ O escalonamento S_a não é serializável

Correto!

☒ O escalonamento S_a não apresenta o problema de atualização perdida

☐ As três transações são equivalentes em conflito

☐ O escalonamento S_a apresenta o problema de leitura não repetitiva

Correto!

☒

Considerando que as últimas operações no escalonamento S_a sejam c_3, c_1, c_2 , nessa ordem, o escalonamento S_a não é recuperável

⋮

Pergunta 2

1,5 / 1,5 pts

Abaixo apresentam-se três transações e um possível escalonamento envolvendo essas transações.

$$T_1 = r(x), r(y), w(x), r(z)$$

$$T_2 = r(z), r(x), r(y), w(z)$$

$$T_3 = r(y), r(z), w(y), r(x)$$

$$S_a = r_3(y), r_2(z), r_1(x), r_2(x), r_3(z), r_2(y), w_3(y), w_2(z), w_1(x), r_3(x), r_1(z)$$

Com base nas transações e nos escalonamento apresentados pode-se afirma que:

Correto!



Considerando a técnica de controle de concorrência por bloqueio compartilhado (ternário) com protocolo 2PL estrito e confirmação (*commit*) implícita (*commit* da transação ocorre logo após a última operação da transação no escalonamento), o escalonamento S_a possui *deadlock*.

Correto!

☒ O *timestamp* da transação T_3 é menor que o *timestamp* da transação T_1 .

☐ As três transações são equivalentes em conflito.

Correto!



Considerando a técnica de controle de concorrência por bloqueio exclusivo (binário) com protocolo 2PL conservador (estático) e confirmação (*commit*) implícita (*commit* da transação ocorre logo após a última operação da transação no escalonamento), o escalonamento S_a não possui *deadlock*.



A técnica de detecção de *deadlock* usando grafo de espera deve ser associada à técnica de controle de concorrência por ordenação de registros de *timestamp* para garantir que o escalonamento S_a execute corretamente, evitando *deadlocks* desnecessários.



Considerando a técnica de controle de concorrência por ordenação de registros de *timestamp*, o escalonamento S_a possui *deadlock*.



Considerando a técnica de controle de concorrência por bloqueio compartilhado (ternário) com protocolo 2PL conservador (estático) e confirmação (*commit*) implícita (*commit* da transação ocorre logo após a última operação da transação no escalonamento), o escalonamento S_a possui *deadlock*.

☐ A transação T_2 é uma transação que apresenta o problema de *starvation* no escalonamento S_a .

Correto!



Considerando a técnica de controle de concorrência por bloqueio exclusivo (binário) com protocolo 2PL estrito e confirmação (*commit*) implícita (*commit* da transação ocorre logo após a última operação da transação no escalonamento), o escalonamento S_a possui *deadlock*.

Pontuação do teste: 3 de 3