

Resolução e Critérios de Correção

U.C. 21103

Sistemas de Gestão de Bases de Dados

19 de fevereiro de 2013

INSTRUÇÕES

- O tempo de duração da prova de p-fólio é de 90 minutos.
- O estudante deverá responder à prova na folha de ponto e preencher o cabeçalho e todos os espaços reservados à sua identificação, com letra legível.
- Visto que o enunciado da prova não é utilizado para resposta, poderá ficar na posse do mesmo.
- Verifique no momento da entrega das folhas de ponto se todas as páginas estão rubricadas pelo vigilante. Caso necessite de mais do que uma folha de ponto, deverá numerá-las no canto superior direito.
- Em hipótese alguma serão aceites folhas de ponto dobradas ou danificadas.
- Exclui-se, para efeitos de classificação, toda e qualquer resposta apresentada em folhas de rascunho.
- Os telemóveis deverão ser desligados durante toda a prova e os objectos pessoais deixados em local próprio da sala de exame.
- O enunciado da prova é constituído por **2** páginas e termina com a palavra **FIM**. Verifique o seu exemplar do enunciado e, caso encontre alguma anomalia, dirija-se ao professor vigilante nos primeiros 15 minutos da mesma, pois qualquer reclamação sobre defeitos de formatação e/ou de impressão que dificultem a leitura não será aceite depois deste período.
- Utilize unicamente tinta azul ou preta.
- O exame é sem consulta. A interpretação das perguntas também faz parte da sua resolução, se encontrar alguma ambiguidade deve indicar claramente como foi resolvida.

A informação da avaliação do estudante está contida no vetor das cotações:

Questão: 1 2 3 4 5

C: 25 25 25 25 20 décimas

Grupo A – Sistemas de Bases de Dados

1. (2,5 valores) Que políticas de substituição das memórias temporárias (“buffer replacement”) conhece? Quais as mais utilizadas em sistemas de bases dados relacionais? Justifique a sua resposta.
(Resposta: 1/2 página)

Resposta:

- As políticas de substituição das memórias temporárias (buffers) pretendem minimizar o acesso a disco. As duas mais conhecidas são: a política LRU (least recently used) e a MRU (*most recently used*).

- Nos sistemas operativos geralmente utilizam a política LRU (*least recently used*) de substituição de blocos, i.e., se um bloco tem de ser substituído vamos remover o que foi utilizado menos recentemente.

- Nas bases de dados a estratégia LRU é uma má estratégia. Por exemplo, na junção de duas tabelas (empréstimo |><| cliente) o algoritmo "nested loop join" contém um ciclo interno e um ciclo externo:

```
01 For each tuple in empréstimo
02   For each tuple in cliente
03     If empréstimo.id_cliente=cliente.id_cliente
04       Begin ..... end
05     End for
06 End for
```

- No caso dos blocos com informação do cliente, depois de tratados, podem ser imediatamente removidos. Assim, para as bases de dados, a substituição dos blocos mais recentemente utilizados é a estratégia mais adequada, i.e., MRU (*most recently used*).

Manual: As políticas de substituição dos buffers, página 462 e seguintes.

Critérios de correção:

- 1,0 valores, referir as políticas
- 1,5 valores, justificação da MRU

2. (2,5 valores) Mostre como deriva uma das seguintes expressões relacionais utilizando regras de equivalência para satisfazer a igualdade: $\sigma_{\theta_1 \wedge \theta_2 \wedge \theta_3}(E) = \sigma_{\theta_1}(\sigma_{\theta_2}(\sigma_{\theta_3}(E)))$.
(Resposta: 1/2 página)

Resposta:

Utilizando as seguintes regras de equivalência:

Regra A: $\sigma_{\theta_1 \wedge \theta_2}(E) = \sigma_{\theta_1}(\sigma_{\theta_2}(E))$ uma operação de seleção conjuntiva pode ser decomposta por uma sequência de seleções individuais;

Podemos encontrar as equivalências com as seguintes justificações:

$\sigma_{\theta_1 \wedge \theta_2 \wedge \theta_3}(E)$	dado
$\sigma_{\theta_1 \wedge \theta_2}(\sigma_{\theta_3}(E))$	aplicando a regra A
$\sigma_{\theta_1}(\sigma_{\theta_2}(\sigma_{\theta_3}(E)))$	aplicando a regra A

Manual: Regras de equivalência, página 571 e seguintes.

Critérios de correção:

- 1,0 valores, derivação das equivalências
- 1,5 valores, justificação das regras

3. (2,5 valores) O que entende por ACID num sistema de base de dados transacional?
Exemplifique falhas das propriedades de atomicidade e consistência.
(Resposta: 1/2 página)

Resposta:

ACID é o acrónimo de quatro propriedades: Atomicidade, Consistência, Isolamento e Durabilidade.

(a) Atomicidade - a transação executa todas as ações ou pelo contrário nenhuma delas, isto é, não existem execuções parciais das transações. A ideia de atomicidade está associada à indivisibilidade; as transações são indivisíveis.

Para a transação T, que realiza uma transferência bancária, subtraindo 100 euros na conta A e adicionando 100 euros na conta B;

0001: T writes $A=A-100$

0002: T writes $B=B+100$

um exemplo de falha da atomicidade será a execução da primeira ação (T writes $A=A-100$) e a não execução da segunda ação (T writes $B=B+100$). Para este tipo de falha o SGDB deverá abortar a transação T.

(b) Consistência - a transação ao utilizar uma base de dados consistente deve deixá-la consistente no final da mesma transação. Sem o requisito da consistência, as quantidades de dinheiro podiam ser criadas ou destruídas numa transação. A responsabilidade da consistência recai sobre o programador da transação.

Para a transação T, que realiza uma transferência bancária, subtraindo 100 euros na conta A e adicionando 100 euros na conta B, um exemplo de falha de consistência seria

0001: T writes $A=A-100$

0002: T writes $B=B+150$

onde foi realizado um aumento de 50 euros sem qualquer explicação.

Critérios de correção:

- 0,5 valores, definição ACID

- 1,0 valores, exemplo falhas atomicidade

- 1,0 valores, exemplo falhas consistência

4. (2,5 valores) Considere a seguinte sequência do "log" para recuperação de uma base de dados. O que é realizado na fase de Redo().

LSN	LOG
00	Begin checkpoint
10	End checkpoint
20	T1 writes P1
30	T2 writes P2
40	T3 writes P3
50	T2 commit
60	T3 writes P2
70	T2 end
80	T1 writes P5
90	T3 abort
	CRASH, RESTART

(Resposta: 1 página)

Resposta:

O *Redo* inicia com o menor LSN da DPT, LSN=20. O *log* é varrido desde LSN 20 até ao ponto de *crash*, voltando a aplicar as alterações (*writes* ou CLR) do *log*.

LSN 20 Redo write P1

LSN 30 Redo write P2

LSN 40 Redo write P3

LSN 50 No action

LSN 60 Redo write P2

LSN 70 No action

LSN 80 Redo write P5

LSN 90 No action

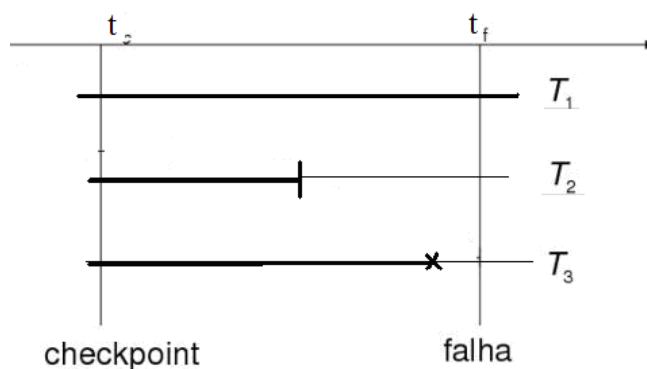
De seguida, o *Undo* é iniciado no maior LSN da TT, LSN=80, o *log* é varrido do fim para o princípio. A instrução de CLR é acrescentada ao *log* para cada ação de desmontagem.

Em conclusão para cada transação:

T1 fez *rollback* (*redo*, *undo*)

T2 fez *commit* (*redo*)

T3 fez *rollback* (*redo*, *undo*)



Critérios de correção:

- 2,0 valores, processo Redo()

- 0,5 valores, compreensão problema

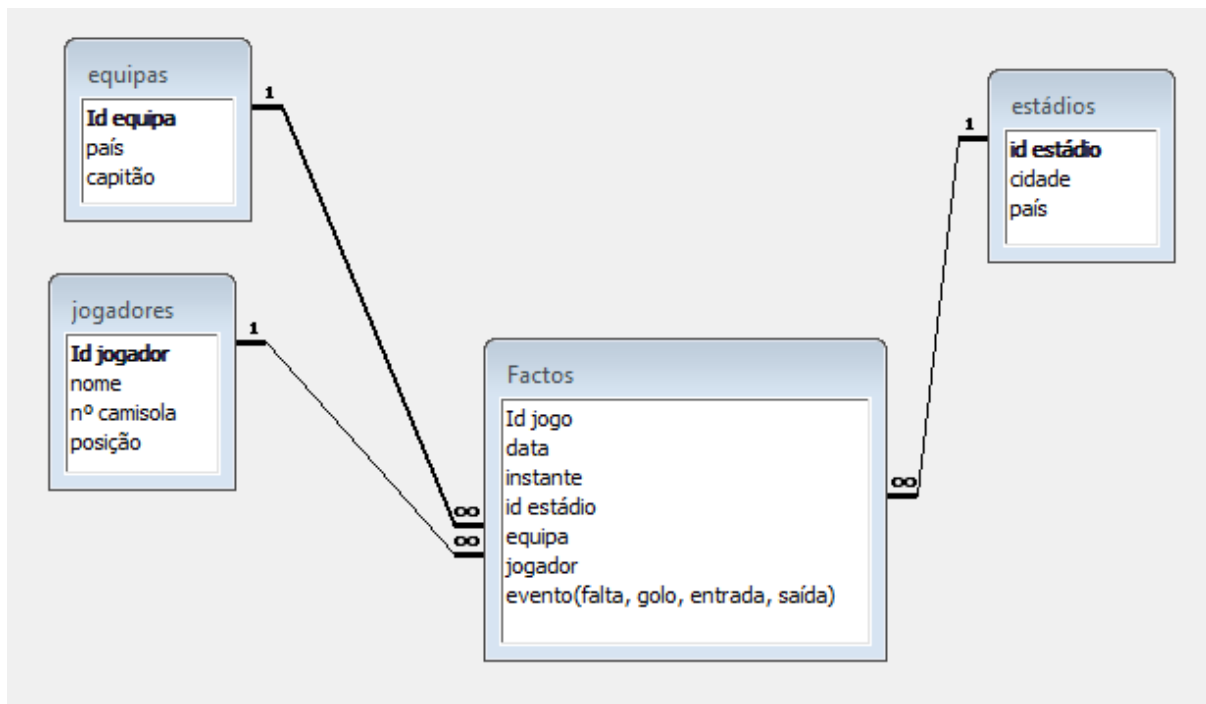
Grupo B – Prática em “Data Warehousing”

5. (2 valores) Pretendemos desenhar um “Data Warehouse” para o seguinte sistema. Defina a tabela de factos em primeiro lugar. De seguida, defina três dimensões para o “Data Warehouse” e apresente a tabela de factos associada às três dimensões.

Considere a seguinte base de dados, que armazena informação sobre o Campeonato Europeu de Futebol e que considera as equipas, os futebolistas, os capitães, os estádios e os jogos:

- Equipas: deve ter para cada uma delas o seu identificador e o país que representam.
- Futebolistas: deve registar o seu identificador, o seu nome, número da camisola, posição em que jogam (e.g. guarda-redes, defesa central, etc...), bem como o identificador da equipa em que jogam.
- Capitães: deve registar os futebolistas que são capitães de uma dada equipa.
- Estádios: dever ter para cada um deles o seu identificador e a cidade onde estão localizados.
- Jogos: deve ter um identificador, bem como o identificador do estádio, data em que decorreram, as equipas envolvidas e do resultado final. As faltas com cartão amarelo e vermelho devem ser também registadas e associadas ao futebolista. Para os golos e as faltas deve ser conhecido o instante do jogo em que ocorreram.

(Resposta: 1 página)



Critérios de correção:

- criar DW com 1 tabelas de factos e 3 dimensões
- penalização até 50% para esquema mal desenhado
- penalização até 50% atributos desadequados na tabela Factos
- penalização até 50% dimensões desadequadas
- penalização até 50% ligações mal estabelecidas

FIM