

MAC426 – Lista de Exercícios: Nível Físico

Marcelo Finger

Exercício 1

- (a) Dê um exemplo de escalonamento (*schedule*) de transações seriável que não poderia ser obtido seguindo-se o protocolo de bloqueio bifásico (*two-phase locking*)
- (b) Mostre que há escalonamentos possíveis no protocolo *two-phase locking* que não são possíveis seguindo-se o protocolo de *timestamps* (marcação de tempo).
- (c) Considere um banco de dados centralizado (isto é, não distribuído) que usa o protocolo de bloqueio bifásico (*two-phase locking*) no qual todas as transações sejam programas sequenciais (portanto nenhuma transação pode ter mais de um “read” ou “write” que esteja bloqueado). É possível que uma transação esteja envolvida em mais que um *deadlock* (travamento)? Justifique sua resposta.

Exercício 2

Considere duas transações T_1 e T_2 , dois registros do banco de dados x e y , e considere o seguinte escalonamento de suas operações:

Transação	Operação
T_1	escreve 1 em x
T_1	escreve 3 em y
T_2	escreve 1 em y
T_1	COMMIT
T_2	lê x

Pergunta-se: Este escalonamento é seriável? Justificar.

Exercício 3

Diz-se que um escalonador pode implementar abortos por *restituição da imagem prévia* se, ao para desfazer uma transação abortada, basta restabelecer o valor dos registros anterior a cada operação de escrita.

Considere o escalonamento da questão anterior acrescido de uma operação de aborto.

<i>Transação</i>	<i>Operação</i>
T_1	escreve 1 em x
T_1	escreve 3 em y
T_2	escreve 1 em y
T_1	COMMIT
T_2	lê x
T_2	ABORT

Este aborto pode ser implementado por restituição da imagem prévia? Justifique.

Exercício 4

Os sistemas operacionais usualmente lidam com o gerenciamento de memória das aplicações, isto é, se no momento da execução de uma operação $\text{input}(X)$ não houver espaço disponível na memória, o sistema operacional escolherá algum bloco de memória B para ser guardado no disco; quando a aplicação futuramente precisar deste bloco B , o bloco é carregado do disco. Do ponto de vista dos esquemas usuais de recuperação de crash em banco de dados, tal política é conveniente ? Por que ?

Exercício 5

Explique por que acoplar o esquema de *checkpointing* a outros mecanismos de recuperação de crash — como log incremental, com ou sem atualizações imediatas — pode tornar a recuperação com tais mecanismos mais eficiente.

Exercício 6

O mecanismo de controle de concorrência em transações por bloqueios bifásicos (two-phase locking, 2PL) é suscetível à ocorrências de travamentos (dead-locks), ou seja, uma situação na qual um conjunto de transações fica impossibilitado de executar qualquer operação de escrita/leitura.

Pede-se:

- Mostra um escalonamento 2PL de um conjunto de transações no qual há ocorrência de um travamento.

- (b) Descrever um mecanismo que evita que as transações travadas permaneçam neste estado indefinidamente. Detalhar:
 - (b₁) as estruturas de dados utilizado por este mecanismo; e
 - (b₂) o funcionamento deste mecanismo.

Este mecanismo não deverá impor restrições à duração permitida às transações em execução.

Exercício 7

Considere as relações $R_1 = (A, B, C)$ e $R_2 = (A, D, E)$. Para cada uma das situações abaixo esboce um algoritmo que compute a junção natural $R_1 \bowtie R_2$ e estime o número de acessos a bloco numa execução desse algoritmo. Seus algoritmos devem explorar tanto quanto possível as estruturas físicas existentes. Descreva-os num nível de abstração bem alto, em português ou pseudo-código de alto nível. Suas estimativas devem ser feitas em termos de parâmetros como o número de blocos de dados de R_i , o número de tuplas de R_i , etc.

- (a) Um índice de hash secundário é mantido sobre o atributo A de R_1 e nenhum índice é mantido sobre o atributo A de R_2 . Assuma que cada “hash bucket” tem comprimento de 1 bloco.
- (b) Duas árvores-B+ são mantidas como índices secundários, uma sobre o atributo A de R_1 , a outra sobre o atributo A de R_2 .
- (c) Estruturas primárias são mantidas sobre ambas as relações, de modo que cada uma delas é fisicamente armazenada em ordem crescente dos valores do atributo A .

Exercício 8

Um sistema S consiste de 3 máquinas M_i ($i = 1, \dots, 3$), cada uma com 4 processadores $P_{i,j}$ ($j = 1, \dots, 4$) e com 2 discos $D_{i,k}$ ($k = 1, \dots, 2$). O sistema operacional permite U usuários ao mesmo tempo, com um máximo de C processos concorrentes. O sistema S é usado para sediar um banco de dados (com propriedades ACID) que aceita até T transações concorrentes (T pode ou não ser maior que C), não tem replicação, e faz controle de concorrência no nível de registro.

- (a) Suponha que o controle de concorrência é por travamento bifásico (*two-phase locking*). Quantas transações podem ser simultaneamente executadas contra o BD?
- (b) Responda novamente à pergunta acima, agora supondo que o controle de concorrência é por ordenação de marcas de tempo (*timestamp ordering*).
- (c) Quantas transações que acessam o mesmo registro r no disco $D_{1,1}$ podem ser simultaneamente executadas contra o BD caso o controle de concorrência seja por travamento bifásico?
- (d) E caso o controle de concorrência seja por ordenação de marcas de tempo?

Exercício 9

Considere as três transações

T_1 : read(X); write(X); read(Y); write(Y)

T_2 : read(Z); read(Y); write(Y); read(X); write(Y)

T_3 : read(Y); read(Z); write(Y); write(Z)

A figura abaixo mostra 2 escalonamentos (A e B) para essas transações.

T_1	T_2	T_3
read(X) read(X)	read(Z) read(Y) read(Y)	read(Y) read(Z)
read(Y) write(Y)	read(X) write(X)	write(Y) write(Z)

Escalonamento A

T_1	T_2	T_3
read(X) write(X)	read(Z)	read(Y) read(Z)
read(Y) write(Y)	read(Y) write(Y) read(X) write(X)	

Escalonamento B

- Determine se os escalonamentos A e B são aceitáveis pelo protocolo do travamento bifásico (*two-phase locking*).
- Determine se eles são aceitáveis pelo esquema de ordenação de marcas de tempo (*timestamp ordering*).
- Idem, pelo esquema multiversão com marcas de tempo (*multi-version timestamping*).
- Determine se esses escalonamentos são seriáveis.

Indique, nos casos (a), (b) e (c), que operações são efetuadas pelo sistema.

Exercício 10

- Explique quais são as dificuldades envolvidas em adaptar a técnica de *shadow paging* (página sombra) para sistemas que permitem a execução concorrente de transações.
- Identifique aplicações ou cenários em que cada uma das técnicas de recuperação abaixo sejam mais apropriadas:
 - *logging* com atualizações imediatas;
 - *logging* com atualizações a posteriori (*deferred updates*);
 - *logging* com *checkpointing*.