



Trabajo Práctico - Transacciones

Ejercicio 1

Durante su ejecución una transacción pasa a través de varios estados hasta que se compromete o aborta. Listar todas las secuencias posibles de estados por los que puede pasar una transacción. Explicar por qué puede ocurrir cada una de las transiciones de estados.

Ejercicio 2

¿Cuáles son los tres problemas de concurrencia en el acceso a datos que pueden suceder cuando se realizan transacciones? Escriba un ejemplo para cada uno de ellos.

Ejercicio 3

- ¿Qué es una planificación de n transacciones?
- ¿Qué es una planificación serie?

Ejercicio 4

Para cada una de las siguientes planificaciones no serie, indicar si es serializable en cuanto a conflictos. Justificar.

Planificación 1

T1	T2	T3	T4
R(A)			
		R(C)	
	R(B)		
W(A)			
			R(D)
		W(C)	
	W(B)		
			W(D)

Planificación 2

T1	T2	T3	T4
	R(A)		
	W(B)		
			R(B)
			W(B)
		R(C)	
		W(B)	
R(A)			
R(B)			
R(D)			
W(A)			
W(D)			

Planificación 3

T1	T2	T3
	C:= 10 write (C)	
		read (C)
		read (A)
		A:= A *C
read (B)		
A:= B + 2		
		write (A)
B:= B * 2		
write (A)		
	A:= 100 write (A)	

Ejercicio 5

Para las siguientes transacciones T_6 y T_7 con $A = B = 25$:

- ¿Cuáles son los resultados posibles con la ejecución en serie de estas transacciones? Justificar.
- Armar dos planificaciones no serie serializables en cuanto a conflictos. Justificar.
- Armar una planificación no serie NO serializable en cuanto a conflictos.

T_6	T_7
Read (A)	Read (A)
$A := A + 100$	$A := A * 2$
Write (A)	Write (A)
Read (B)	Read (B)
$B := B + 100$	$B := B * 2$
Write (B)	Write (B)

Ejercicio 6

Sean las transacciones T_8 , T_9 y T_{10} definidas de la siguiente forma:

T_8	T_9	T_{10}
Read (A)	Read (A)	Read (A)
$A := A + 1$	$A := A * 2$	Mostrar (A)
Write (A)	Write (A)	$A := 1$

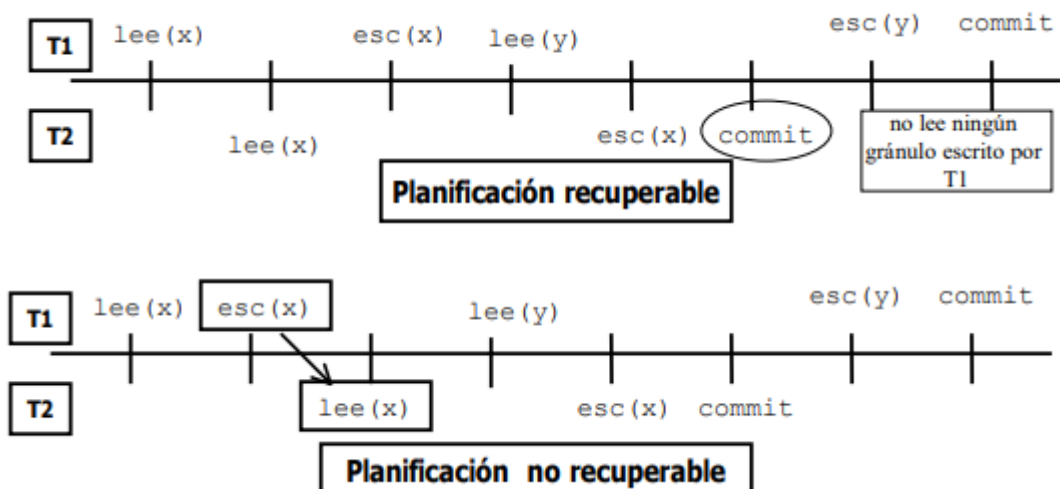
Suponiendo que las transacciones T_8 , T_9 y T_{10} se pueden ejecutar concurrentemente y que inicialmente $A=0$, ¿cuántos y cuáles resultados correctos posibles se pueden tener?

Ejercicio 7

Indicar si cada una de las siguientes planificaciones es recuperable o no.

Recordar:

Una planificación P es recuperable si ninguna transacción T de P se confirma antes de que se hayan confirmado todas las transacciones T' que han escrito un elemento que T lee posteriormente.



a)

T1	T2
R(A)	
	R(B)
	W(A)
W(B)	
	commit
commit	

b)

T1	T2	T3
		W(B)
	R(A)	
R(B)		
		abort
	W(B)	
W(A)		
commit		
	commit	

c)

T1	T2	T3	T4
R(A)			
		R(B)	
	R(C)		
		W(C)	
R(B)			
			R(B)
W(C)			
	R(D)		
			R(D)
		W(C)	
			W(B)
	W(A)		
commit			
		commit	
	abort		
			commit

Ejercicio 8

Considere las transacciones T_1 , T_2 y T_3 y las Planificaciones P_1 y P_2 que aparecen a continuación. Dibujar los grafos de precedencia para P_1 y P_2 y explique si cada planificación es serializable o no. Para aquellas planificaciones serializables indicar la planificación serie equivalente. Justificar

T_1 : $r_1(X)$; $r_1(Z)$; $w_1(X)$

T_2 : $r_2(Z)$; $r_2(Y)$; $w_2(Z)$; $w_2(Y)$

T_3 : $r_3(X)$; $r_3(Y)$; $w_3(Y)$

P_1 : $r_1(X)$; $r_2(Z)$; $r_1(Z)$; $r_3(X)$; $r_3(Y)$; $w_1(X)$; $w_3(Y)$; $r_2(Y)$; $w_2(Z)$; $w_2(Y)$

P_2 : $r_1(X)$; $r_2(Z)$; $r_3(X)$; $r_1(Z)$; $r_2(Y)$; $r_3(Y)$; $w_1(X)$; $w_2(Z)$; $w_3(Y)$; $w_2(Y)$

Ejercicio 9

a) ¿El protocolo de bloqueo garantiza planificaciones serializables? Justificar.

b) ¿Qué ventajas y desventajas proporciona la alternativa de bloqueo de dos fases estricto?

Ejercicio 10

Para cada conjunto de transacciones, construir una planificación no serie usando bloqueo de dos fases estricto. Interpolan las instrucciones entre todas las transacciones participantes en cada planificación.

a) $T_1 = R(A), R(B), W(B), W(C)$

$T_2 = R(D), R(E), W(E), W(D)$

$T_3 = R(B), R(D), W(B), W(D)$

$T_4 = R(A), W(B), R(C), W(D)$

b) $T_1 = R(A), W(A), W(B), R(C), W(C)$

$T_2 = R(C), W(C)$

$T_3 = R(B), W(B), W(C), R(D), W(D)$

Ejercicio 11

Para la siguiente planificación, indicar si cumple con el protocolo de bloqueo de dos fases. Justificar

$P = BC_1[A] BC_4[B] LB_1[A] BC_5[B] BC_2[A] BE_1[C] LB_2[A] LB_4[B] LB_1[C] BE_3[A] LB_5[B] LB_3[A] BC_5[D] BE_3[B] LB_5[D] BC_4[D] LB_4[D] LB_3[B] BE_6[A] LB_6[A] BE_7[D] LB_7[D] BE_7[A] LB_7[A]$.

Obs: El subíndice indica el número de transacción.

Ejercicio 12

Para cada conjunto de transacciones, construir una planificación no serie usando bloqueo de dos fases básico de manera de generar interbloqueos. Justificar

a) $T_1 = R(A), W(D)$

$T_2 = R(A), W(C)$

$T_3 = R(D), W(A)$

$T_4 = R(C), W(D)$

b) $T_1 = R(A), R(D), R(C), W(D)$

$T_2 = R(A), R(B), R(E), W(E)$

$T_3 = R(D), R(E), R(A), W(F), W(A)$

Ejercicio 13

Para las transacciones T_1 y T_2 que se muestran a continuación, con valor inicial de $A = 100$, $B = 200$, $C = 300$ y $D = 400$.

$T_1 = \text{Read}(B); B = B * 4; \text{Write}(B); \text{Read}(C); \text{Read}(D); C = B + D; \text{Write}(C)$

$T_2 = \text{Read}(A); A = A + 100; \text{Write}(A); \text{Read}(B); \text{Read}(C); B = C; \text{Write}(B)$

a) Dar una planificación concurrente usando el protocolo de bloqueos de dos fases que no caiga en deadlock. ¿Cuáles son los valores finales de A , B , C y D ?

b) Dar una planificación concurrente usando el protocolo de bloqueos de dos fases que caiga en deadlock. Indicar claramente el punto donde se produce el deadlock.

Ejercicio 14

a) A partir de los siguientes registros de bitácora (con modificación diferida), indicar cuáles transacciones se ignoran y cuáles se rehacen.

< T_1 comienza>

< T_1 , D , 20>

< T_1 finaliza>

<punto de verificación>

< T_4 comienza>

< T_4 , B , 15>

< T_4 , A , 20>

< T_4 finaliza>

< T_2 comienza>

< T_2 , B , 12>

< T_3 comienza>

< T_3 , A , 30>

< T_2 , D , 25> "

caída del sistema

b) ¿Qué modificaciones deberían hacerse si fuese bitácora con modificación inmediata? ¿Qué acciones se tomarían entonces con las transacciones?

Ejercicio 15

Suponga que utilizamos el protocolo de actualización inmediata, construir la bitácora para la siguiente planificación y describa el proceso de recuperación ante la caída del sistema.

