

2º Parcial Diseño y Gestión de Bases de Datos (14/01/2025)

Apellidos, Nombre:

Contesta las preguntas del examen con claridad y brevedad el espacio disponible.

1. Explica por qué es necesario el uso de transacciones. (0,1 puntos)

Porque las transacciones nos van a permitir posponer la comprobación de algunas restricciones hasta el final de las mismas, lo que nos va a permitir realizar operaciones que de otra forma no se podrían realizar.

2. a) 0,3 puntos b) 0,1 puntos c) 0,1 puntos

Dadas las siguientes definiciones de relaciones

```
CREATE TABLE R
( A CHAR(3) CONSTRAINT PK_R PRIMARY KEY,
  B VARCHAR2(10), C INTEGER);
CREATE TABLE S
( D INTEGER CONSTRAINT PK_S PRIMARY KEY,
  E VARCHAR2(10),
  F CHAR(3) CONSTRAINT FK_S_R REFERENCES R(A)
  DEFERRABLE INITIALLY IMMEDIATE);
CREATE TABLE T
( A CHAR(3) CONSTRAINT PK_T PRIMARY KEY,
  G INTEGER, H CHAR(3) CONSTRAINT FK_T_R REFERENCES R(A)
  DEFERRABLE INITIALLY IMMEDIATE);
```

- a) Escribe con instrucciones SQL la transacción que permitiría cambiar a 'XXX' el valor del atributo A de la fila de R que cumple A='YYY' independientemente del contenido de la base de datos.

```
SET CONSTRAINT FK_S_R DEFERRED;
SET CONSTRAINT FK_T_R DEFERRED;
UPDATE R SET A='XXX' WHERE A='YYY';
UPDATE S SET F='XXX' WHERE F='YYY';
UPDATE T SET H='XXX' WHERE H='YYY';
COMMIT;
```

- b) En la transacción definida en el apartado a) , ¿en qué momento se comprueba cada una de las restricciones afectadas?

La restricción PK_R se comprobará tras la operación de modificación en R y las restricciones FK_S_R y FK_T_R se comprobarían al terminar la transacción (después del COMMIT).

- c) Si en la definición de las relaciones no se hubiera indicado el modo de comprobación de las restricciones, ¿cómo afectaría a la modificación planteada en el apartado a)?

La modificación del atributo A de R sólo se podrá realizar cuando no haya filas en S y T que hagan referencia a la fila modificada.

3. a) 0,3 puntos b) 0,1 puntos c) 0,1 puntos d) 0,1 puntos

Dadas dos transacciones T1 y T2, y un elemento de datos X cuyo último valor confirmado es 10. Asumiendo que inicialmente los buffers de datos están vacíos, resuelve los siguientes apartados:

- a) Rellena la tabla que se muestra a continuación, indicando para cada una de las estrategias de actualización, qué valor contendría el elemento X en el buffer de datos y en disco en cada instante de tiempo (en caso de que haya más de un valor posible indícalos todos). En cada momento se debe mostrar el valor (o valores) después de ejecutar la operación (el subíndice de la operación indica la transacción que la realiza).

TIEMPO	OPERACIÓN	Inmediata-No Forzar		Inmediata-Forzar		Diferida-No Forzar		Diferida-Forzar	
		buffer	disco	buffer	disco	buffer	disco	buffer	disco
t ₀	r ₁ (X)	10	10	10	10	10	10	10	10
t ₁	X=X+3	10	10	10	10	10	10	10	10
t ₂	w ₁ (X)	13	10/13	13	10/13	13	10	13	10
t ₃	c ₁	13	10/13	13	13	13	10/13	13	13
t ₄	r ₂ (X)	13	10/13	13	13	13	10/13	13	13
t ₅	X=7	13	10/13	13	13	13	10/13	13	13
t ₆	w ₂ (X)	7	10/13/7	7	13/7	7	10/13	7	13

b) ¿Se produce alguna escritura del diario en disco? Si es así, explica cuándo sucede.

Cuando los valores de X actualizados en t2 y t6 se lleven a disco y en la confirmación de la transacción T1

c) Si en el instante t_7 se produce un fallo del sistema con pérdida de MP, indica qué tarea de recuperación se ejecutaría para cada transacción en cada una de las estrategias.

		Inmediata-No Forzar		Inmediata-Forzar		Diferida-No Forzar		Diferida-Forzar	
TIEMPO	OPERACIÓN	T1	T2	T1	T2	T1	T2	T1	T2
t_7	fallo	Rehacer	Deshacer	Nada	Deshacer	Rehacer	Nada	Nada	Nada

d) Indica qué valores de X contendrían el buffer y el disco después de la recuperación en cada una de las estrategias.

	Inmediata-No Forzar		Inmediata-Forzar		Diferida-No Forzar		Diferida-Forzar	
TIEMPO	buffer	disco	buffer	disco	buffer	disco	buffer	disco
t_8		13		13		13		13

4. a) 0,2 puntos b) 0,1 puntos c) 0,3 puntos d) 0,3 puntos e) 0,4 puntos

Sean T1 y T2 las siguientes transacciones:

T1= $r_1(X)$, $w_1(X)$, $w_1(Y)$, c_1

T2= $r_2(X)$, $w_2(Y)$, $w_2(X)$, c_2

a) Escribe una ordenación de las operaciones de T1 y T2 que cumplan lo especificado a continuación. Las ordenaciones deben ser distintas en cada caso:

- La ordenación no constituye un plan:
 $r_1(X)$, $w_1(Y)$, $w_1(X)$, $r_2(X)$, $w_2(Y)$, $w_2(X)$, c_2 , c_1
- La ordenación constituye un plan en serie:
 $r_1(X)$, $w_1(X)$, $w_1(Y)$, c_1 , $r_2(X)$, $w_2(Y)$, $w_2(X)$, c_2
- La ordenación constituye un plan concurrente no serializable por conflictos:
 $r_1(X)$, $w_1(X)$, $r_2(X)$, $w_2(Y)$, $w_1(Y)$, $w_2(X)$, c_1 , c_2
- La ordenación constituye un plan concurrente serializable por conflictos:
 $r_1(X)$, $w_1(X)$, $r_2(X)$, $w_1(Y)$, $w_2(Y)$, $w_2(X)$, c_1 , c_2

b) Escribe un plan de T1 y T2 en el que se produzca una anomalía. Explica la anomalía.

$r_1(X)$, $w_1(X)$, $r_2(X)$, $w_1(Y)$, $w_2(Y)$, $w_2(X)$, c_1 , c_2

En este plan se da la anomalía de lectura sucia ya que T2 lee un valor de X escrito por T1 antes de que T1 se confirme

c) Plantea en la siguiente tabla un plan concurrente de las transacciones T1 y T2 que sea admitido por el protocolo de bloqueo B2F explícito (pueden sobarte filas).

Tiempo	T1	T2
t0	BE(X)	
t1	$r_1(X)$	
t2	$w_1(X)$	
t3	BE(Y)	
t4	D(X)	
t5		BE(X)
t6		$r_2(X)$
t7	$w_1(Y)$	
t8	D(Y)	
t9		BE(Y)
t10		$w_2(Y)$
t11	c_1	
t12		$w_2(X)$
t13		D(X)
t14		D(Y)
t15		c_2

d) ¿Existiría algún plan para T1 y T2 controlado por el protocolo B2F Implícito cuyas dos primeras operaciones fueran $r_1(X)$ y $r_2(X)$ en cualquier orden? En caso afirmativo escribe el plan completo, en caso negativo justifica la respuesta.

No porque en esos planes siempre aparecerá el problema del bloqueo mortal. Si se ejecutan esas dos operaciones en cualquiera de los dos órdenes posibles, ambas transacciones habrán conseguido el bloqueo de lectura sobre X por tanto el bloqueo de escritura sobre X necesario para las operaciones $w_1(X)$ y $w_2(X)$ no lo podrán conseguir ni T1 ni T2 quedando ambas a la espera de que la otra libere el bloqueo lo que sólo ocurre al ejecutar la confirmación.

e) Escribe dos transacciones T1 y T2 y un plan concurrente para sus operaciones que NO sea aceptado por el protocolo de ordenación de marcas de tiempo (OMT) pero Sí sea aceptado por el protocolo multiversión (MV). Indica las hipótesis de partida y rellena las tablas de abajo para justificar la respuesta (pueden sobarte filas y/o columnas). Justifica también el motivo por el que el plan no es aceptado por OMT.

Para facilitar la respuesta a esta pregunta ten en cuenta lo siguiente:

- Las transacciones como mucho puede tener 4 operaciones contando las confirmaciones.
- Sólo pueden aparecer dos elementos de datos (X e Y).
- La notación para las operaciones de las transacciones debe ser la misma que la usada en el apartado a)

T1: $r_1(Y)$, $r_1(X)$, c_1
T2: $r_2(X)$, $w_2(X)$, c_2
Plan:

Tiempo	T1	T2
t_0	$r_1(Y)$	
t_1		$r_2(X)$
t_2		$w_2(X)$
t_3	$r_1(X)$	
t_4	c_1	
t_5		c_2

Plan en serie cronológico T1->T2
 $MT(T1) = t_0$ $MT(T2) = t_1$

PROTOCOLO OMT

Hipótesis: Supongamos que X e Y existen y que las marcas de lectura y escritura de ambas son anteriores al inicio del plan ($MT_L(X) = t_{-9}$, $MT_E(X) = t_{-9}$ $MT_L(Y) = t_{-9}$ $MT_E(Y) = t_{-9}$).

Protocolo OMT					
tiempo	operación	$MT_L(X)$	$MT_E(X)$	$MT_L(Y)$	$MT_E(Y)$
		t_{-9}	t_{-9}	t_{-9}	t_{-9}
t_0	$r_1(Y)$	t_{-9}	t_{-9}	t_0	t_{-9}
t_1	$r_2(X)$	t_1	t_{-9}	t_0	t_{-9}
t_2	$w_2(X)$	t_1	t_1	t_0	t_{-9}
t_3	$r_1(X)$				

Para que T1 pueda leer X en t_3 se debe cumplir que $MT_E(X) \leq MT(T1)$, en este caso no se cumple la desigualdad ya que $t_1 > t_0$. EL problema es que X ha sido escrito por una transacción más joven que T1 por lo que la operación $r_1(X)$ se está realizando después de la operación $w_2(X)$ que es el orden contrario al que se haría en el plan en serie cronológico.

PROTOCOLO MV

Hipótesis: Supongamos que existe una versión de X llamada X_0 y una versión de Y llamada Y_0 y que las marcas de lectura y escritura de ambas versiones son anteriores al inicio del plan ($MT_L(X_0) = t_{-9}$, $MT_E(X_0) = t_{-9}$ $MT_L(Y_0) = t_{-9}$ $MT_E(Y_0) = t_{-9}$).

Protocolo MV							
		X_0		X_1		Y_0	
tiempo	operación	$MT_L(X_0)$	$MT_E(X_0)$	$MT_L(X_1)$	$MT_E(X_1)$	$MT_L(Y_0)$	$MT_E(Y_0)$
		t_{-9}	t_{-9}			t_{-9}	t_{-9}
t_0	$r_1(Y)$	t_{-9}	t_{-9}			t_0	t_{-9}
t_1	$r_2(X)$	t_1	t_{-9}			t_0	t_{-9}
t_2	$w_2(X)$	t_1	t_{-9}	t_1	t_1	t_0	t_{-9}
t_3	$r_1(X)$	t_1	t_{-9}	t_1	t_1	t_0	t_{-9}
t_4	c_1	t_1	t_{-9}	t_1	t_1	t_0	t_{-9}
t_5	c_2	t_1	t_{-9}	t_1	t_1	t_0	t_{-9}

En t_3 hay dos versiones de X:

- X_0 : creada por una transacción más vieja que T1 (concretamente una transacción cuya marca de tiempo es t_{-9})
- X_1 : creada por una transacción más joven que T1 (concretamente una transacción cuya marca de tiempo es t_1 , es decir T2)

de esas dos versiones T1 lee X_0 pero no actualiza la marca de lectura de esta versión ya que el valor que tiene (t_1) es mayor que la marca de tiempo de T1 (t_0).