

Guía de estudio para el examen

Transacciones

Problema 1: Suma Invariante

Sean tres atributos A, B y C. Indique, para cada par de transacciones $T1$ y $T2$, si se garantiza que $A+B+C$ no cambia.

1)	T1	T2
	Lock A;	Lock B;
	$A = A - 10;$	$B = B - 20;$
	Unlock A;	Unlock B;
	$B = B + 10;$	$C = C + 20;$

2)	T1	T2
	$A = A - 10;$	Lock B;
	Lock B;	$B = B - 20;$
	$B = B + 10;$	Unlock B;
	Unlock B;	$C = C + 20;$

3)	T1	T2
	Lock A;	Lock A;
	$A = A - 10;$	$B = B - 20;$
	Unlock A;	Unlock A;
	$B = B + 10;$	$C = C + 20;$

Problema 2: Bloqueos y caídas

Suponga que varias transacciones han bloqueado distintos elementos de la base de datos. Súbitamente, el computador con la base de datos se apaga debido a problemas de tensión. ¿Qué ocurre con los bloqueos cuando el sistema de base de datos hace rollback?

Problema 3: Deadlock y dos fases

Diseñe un par de transacciones que en una ejecución lleguen a deadlock (abrazo mortal). Rediseñe las transacciones para que cumplan con el protocolo de dos fases (lock-T-unlock). ¿Por qué el deadlock no es posible? Demuestre formalmente la imposibilidad de deadlock en dos fases. ¿Cuál es el trade-off del protocolo de dos fases?

Problema 4: Serialización

Sea el siguiente plan de acción:

<u>T1</u>	<u>T2</u>	<u>T3</u>	<u>T4</u>	<u>T5</u>
Leer A;				
			Leer A;	
	Escribir B;			
				Leer C;
		Escribir A;		
		Escribir B;		
	Leer C;			Escribir B;
	Escribir C;			
			Escribir D;	
Leer D;				

¿Qué conflictos pueden ocurrir en esta agenda? (Entre operaciones)

¿Es serializable por conflicto?

Problema 5: Analogía

Está a punto de ocurrir la principal carrera de caballos del año. Las apuestas son cuantiosas y el hipódromo se está llenando de gente.

Dos individuos, *El Pelusa* y *El Culebra*, preparan el siguiente plan:

1. *El Pelusa* entrará al establo donde están los caballos y dará calmantes a todos los caballos, salvo al *Super Veloz*.
2. Al mismo tiempo, *El Culebra* apostará al *Super Veloz* todo el dinero reunido con el *Pelusa*.
3. Al terminar la carrera, *El Culebra* cobrará las ganancias recaudadas por tan sucia treta de apuestas.

Note que *El Pelusa* deberá desafiar la seguridad del establo para dar los calmantes a los caballos.

¿Qué problemas tiene el plan? ¿A qué concepto hace alusión?

Problema 6: Simultaneidad

Sean T1, T2 y T3 las tres transacciones que se definen a continuación:

T1:

```
UPDATE cuentas SET balance=balance*1.02 WHERE tipo='ahorros';  
UPDATE cuentas SET balance=balance*1.01 WHERE tipo='salario' AND balance<0;  
UPDATE cuentas SET balance=balance*1.07 WHERE tipo='salario' AND balance>0;
```

T2:

```
UPDATE cuentas SET tipo='salario' WHERE nro=12345;
```

T3:

```
UPDATE cuentas SET balance=balance-1000 WHERE nro=12345;
```

Suponga que T1, T2 y T3 se ejecutan de forma más o menos simultánea a nivel de aislamiento READ COMMITTED. ¿Qué cambia respecto al nivel SERIALIZABLE?

Soluciones:

P1: Sólo la 2.

P2: El bloqueo se debe levantar en el rollback. Y no tiene importancia durante la reconstrucción del estado anterior de la BD.

P3: Propuesto.

P4: Ej, T1:Leer A; tiene conflicto con T3:Escribir A; (lo que leyó T1 pierde significado, este es un conflicto *lectura-escritura*). Otro caso ocurre entre T2:Escribir B; y T3:Escribir B; (conflicto *escritura-escritura*). Del último tipo, T2:Escribir D; tiene conflicto con T1:Leer D; (conflicto *escritura-lectura*). En total hay 8 conflictos, aquí sólo se mencionaron algunos). Y no es serializable por conflicto (¿por qué?).

P5: El Culebra espera que El Pelusa logre su cometido. Pero si es atrapado, no se enterará y arriesgará todo el dinero apostado. Esto hace alusión a serializabilidad por conflicto.

P6: Propuesto. (Se les obliga a estudiar BIEN la materia.)

Normalización

Problema 1: Preservación de dependencias

Sea el esquema $R(A,B,C,D)$ y las dependencias funcionales:

1. $B \rightarrow C$
2. $C \rightarrow A$
3. $B \rightarrow D$

¿Cuáles de las siguientes descomposiciones preservan dependencias?

1. $R_1(C,A)$ & $R_2(C,B,D)$
2. $R_1(A,C,D)$ & $R_2(B,D)$
3. $R_1(C,A)$ & $R_2(A,B,D)$

Problema 2: Preservación de dependencias

Sea $R(A,B,C,D,E)$. ¿Cuáles de las siguientes dependencias se puede deducir del resto? (Es redundante)

1. $B \rightarrow C$
2. $A \rightarrow B$
3. $AD \rightarrow C$
4. $DB \rightarrow E$

Problema 3: Normalización en 3FN

Normalice el problema anterior para cumplir 3FN. Remueva cualquier dependencia redundante. Ojo: la clausura de las dependencias no debe cambiar.

Problema 4: Equivalencias

Demuestre que $FNBC \Rightarrow 3FN \Rightarrow 2FN$.

SQL

Problema 1: Películas

Sean los esquemas:

Movie(title, year, length, inColor, studioName, producer)

StarsIn(movietitle, movieyear, starname)

MovieStar(name, address, gender, birthdate)

Sean, también, las siguientes consultas:

Q1:

```
SELECT DISTINCT title, studioName
FROM Movie, StarsIn
WHERE starname='Meryl Streep' AND
title=movietitle AND
year=movieyear;
```

Q2:

```
SELECT DISTINCT title, studioName
FROM Movie, StarsIn, (SELECT starname
FROM StarsIn
HAVING count(*)>10
GROUP BY starname) Productive
WHERE title=movietitle AND year=movieyear AND
Productive.starname=StarsIn.starname;
```

Responda las siguientes preguntas:

1. ¿Qué hacen las consultas Q1 y Q2? Explique brevemente.
2. Convierta las consultas a álgebra relacional. EVITE EL HAVING, pero puede usar la función COUNT(*) junto a GROUP BY.
3. Escriba Q2 en álgebra relacional sin usar COUNT, HAVING ni GROUP BY.