

Segundo Parcial de Fundamentos de Bases de Datos

Noviembre 2016

Presentar la resolución del parcial:

- Duración: 3 horas
- Con las hojas numeradas y escritas de un solo lado.
- Con la cantidad de hojas entregadas en la primer hoja.
- Con cédula de identidad y nombre en cada hoja.
- **Escrita a lápiz y en forma prolija.**

Ejercicio 1 (15 puntos)

Considere el esquema relación $R(A,B,C,D,E,G)$, los conjuntos de dependencias funcionales F y J que se detallan a continuación, y el subesquema de R : $R_1(A,B,C,D)$.

$$F = \{ AB \rightarrow C, CD \rightarrow AG, G \rightarrow B, E \rightarrow D, BD \rightarrow E \}$$

$$J = \{ AB \rightarrow C, CD \rightarrow AGE, G \rightarrow B, E \rightarrow D, BD \rightarrow E, ABE \rightarrow G \}$$

Determinar si las siguientes afirmaciones son verdaderas o falsas justificando cada respuesta.

1. F y J son equivalentes. **VERDADERA.**

$$F \text{ y } J \text{ son equivalentes} \Leftrightarrow F^+ = J^+ \Leftrightarrow F \subseteq J^+ \text{ y } J \subseteq F^+.$$

$$F - J = \{ CD \rightarrow AG \} \quad J - F = \{ CD \rightarrow AGE, ABE \rightarrow G \}$$

$$(CD)^+_J = \{ C,D,A,G,E,B \}, AG \subseteq (CD)^+_J, \text{ por lo tanto } CD \rightarrow AG \in J^+, \text{ entonces } F \subseteq J^+.$$

$$(CD)^+_F = \{ C,D,A,G,B,E \}, AG \subseteq (CD)^+_F, \text{ por lo tanto } CD \rightarrow AG \in F^+, \text{ entonces } J \subseteq F^+.$$

Por lo que se cumple la definición de equivalentes.

2. Todas las claves de R según F tienen por lo menos 2 atributos. **VERDADERA**

Los únicos atributos de R que aparecen solos a la izquierda de alguna dependencia funcional de F son E y G , por lo tanto son los únicos candidatos a ser clave de un atributo. Calculamos sus clausuras para ver si contienen a todos los atributos de R .

$$E^+_F = \{ E, D \} \neq R$$

$$G^+_F = \{ G, B \} \neq R$$

Por lo tanto, no hay claves con un único atributo.

3. ABE es clave de R según F . **VERDADERA**

$$(ABE)^+_F = \{ A, B, E, C, D, G \} = R \quad \underline{ABE} \text{ es superclave}$$

$$A^+_F \neq R, B^+_F \neq R, C^+_F \neq R \text{ (Calculado en la parte anterior).}$$

$$(AB)^+_F = \{A,B,C\} \neq R$$

$$(AE)^+_F = \{A,E,D\} \neq R$$

$$(BE)^+_F = \{B,E,D\} \neq R$$

por lo tanto ABE es superclave minimal (no se puede eliminar ningún atributo y que siga siendo superclave), ABE es clave.

4. Existe un conjunto $X \subseteq R$ tal que X es clave de R según F pero no es clave de R según J. **FALSA**

La noción de clave depende únicamente del esquema y de las dependencias funcionales que se cumplen en él.

En este caso se considera un esquema (R) y dos conjuntos de dependencias equivalentes, por lo tanto las dependencias que se deducen a partir de un conjunto o otro SON LAS MISMAS. Entonces es imposible tener un conjunto de atributos que sea clave según un conjunto de dependencias y que no lo sea con el otro.

5. La máxima forma normal que cumple R1 según F es 2NF. **FALSA**

R1(A,B,C,D)

Es necesario calcular la proyección de las dependencias de F en R1.

$AB \rightarrow C$ y $CD \rightarrow A$ se proyectan directamente.

También se proyecta $CD \rightarrow B$ (ver cálculo de $(CD)^+_F$ en parte 1).

$$(AC)^+_F = \{A,C\}$$

$$(AD)^+_F = \{A,D\}$$

$$(BC)^+_F = \{B,C\}$$

$$(BD)^+_F = \{B,D\}$$

Por lo tanto, las dependencias que se cumplen en R1 son: $\{AB \rightarrow C, CD \rightarrow AB\}$

CD es clave de R1, $CD \rightarrow AB$ cumple las condiciones de BCNF.

AB no es superclave de R1 pero C es primo, $AB \rightarrow C$ cumple las condiciones de 3NF.

La máxima forma normal de R1 es 3NF.

Otra solución:

Por lo calculado en 1) sabemos que $(CD)^+_F = \{C,D,A,G,B,E\}$, $R1 \times (CD)^+_F \setminus$

Por lo calculado en 2) sabemos que:

$$D^+_F = \{D\} \neq R1$$

$$C^+_F = \{C\} \neq R$$

por lo tanto **CD es clave de R1**.

$$(ABD)^+_F = \{A,B,D,C,G,E\}$$
, $R1 \times (ABD)^+_F$

Por lo calculado en en 1) sabemos que:

$$A^+_F = \{A\} \neq R1$$

$$B^+_F = \{B\} \neq R1$$

$$D^+_F = \{D\} \neq R1$$

Por lo calculado en 3) sabemos que:

$$(AB)^+_F = \{A,B,C\} \neq R1$$

$$(AD)^+_F = \{A,D\} \neq R1$$

$$(BD)^+_F = \{B,D\} \neq R1$$

Por lo tanto **ABD es clave de R1.**

A partir de las claves de R1 calculadas podemos afirmar que **A,B,C,D son atributos primos**. Entonces en R1 no se pueden proyectar dependencias que en el lado derecho no tengan un atributo primo, entonces el esquema está por lo menos en 3NF. (Para no cumplirlo tendría que existir una dependencia transitiva de clave a un atributo no primo).

6. R1 cumple 2NF según J. **VERDADERA**

La FN cumple un esquema están determinadas por las dependencias que se cumplen en él. Al ser F y J equivalentes entonces el esquema R1 cumple las mismas FN según cada uno de estos conjuntos de dependencias.

Por la parte anterior, la máxima FN que satisface R1 según F (y por lo tanto J) es 3NF. Por las definiciones de FN se sabe que también cumple 2NF.

Ejercicio 2 (20 puntos)

Dado el esquema relación $R(A,B,C,D,E,G)$, el conjunto de dependencias funcionales F :

$$F = \{ A \rightarrow BC, C \rightarrow DG, BD \rightarrow E, AB \rightarrow D, BC \rightarrow G \}$$

y la descomposición de R ρ : $(R1(A,B,C), R2(C,D,G), R3(B,D,E))$.

1. Determinar en que forma normal se encuentra R respecto de F . Justificar la respuesta.

A no pertenece a la derecha de las dependencias funcionales por lo tanto pertenece a todas las claves.

$$A^+ = \{A,B,C,D,G,E\} = R$$

A es la única clave de R según F .

En $C \rightarrow DG$

C no es superclave, $C \rightarrow DG$ viola BCNF.

D,G no son atributos primos, $C \rightarrow DG$ viola 3NF

Como la única clave es A (un único atributo) no hay dependencias parciales de claves a atributos no primos.

R está en 2NF.

2. Determinar si la descomposición dada es con JSP. Justificar la respuesta.

Se aplica el test para determinar si una descomposición es con JSP.

	A	B	C	D	E	G
R1(ABC)	a ₁	a ₂	a ₃	b ₁₄	b ₁₅	b ₁₆
R2(CDG)	b ₂₁	b ₂₂	a ₃	a ₄	b ₂₅	a ₆
R3(BDE)	b ₃₁	a ₂	b ₃₃	a ₄	a ₅	b ₃₆

A partir de la dependencia $C \rightarrow DG$, se igualan los atributos D,G en la primera y segunda fila.

	A	B	C	D	E	G
R1(ABC)	a ₁	a ₂	a ₃	a ₄	b ₁₅	a ₆
R2(CDG)	b ₂₁	b ₂₂	a ₃	a ₄	b ₂₅	a ₆
R3(BDE)	b ₃₁	a ₂	b ₃₃	a ₄	a ₅	b ₃₆

A partir de la dependencia $BD \rightarrow E$, se igualan los atributos E en la primera y tercer fila.

	A	B	C	D	E	G
R1(ABC)	a ₁	a ₂	a ₃	a ₄	a ₅	a ₆
R2(CDG)	b ₂₁	b ₂₂	a ₃	a ₄	b ₂₅	a ₆
R3(BDE)	b ₃₁	a ₂	b ₃₃	a ₄	a ₅	b ₃₆

Se completó una fila de todos a, por lo tanto la descomposición es con JSP.

3. Determinar en que forma normal se encuentra ρ . Justificar la respuesta.

$R1(A,B,C)$

$A \rightarrow BC$ se proyecta directamente.

$B^+ = \{B\}$

$C^+ = \{C,D,G\}$

$(BC)^+ = \{B,C,D,G,E\}$

La única dependencia que se proyecta en $R1$ es: $A \rightarrow BC$

$R2(C,D,G)$

$C \rightarrow DG$ se proyecta directamente.

$D^+ = \{D\}$

$G^+ = \{G\}$

$(DG)^+ = \{D,G\}$

La única dependencia que se proyecta en $R2$ es: $C \rightarrow DG$

$R3(B,D,E)$

$BD \rightarrow E$ se proyecta directamente.

$E^+ = \{E\}$

$(DE)^+ = \{D,E\}$

La única dependencia que se proyecta en $R3$ es: $BD \rightarrow E$

En todos los casos los lados izquierdos de las dependencias funcionales son superclaves de sus respectivos esquemas, por lo tanto cumplen la condición de BCNF.

En los esquemas no se cumplen dependencias multivaluadas no triviales, no funcionales por lo tanto los esquemas cumplen la condición de 4NF.

Por lo tanto la descomposición se encuentra en 4NF.

4. Determinar si ρ preserva las dependencias funcionales. Justificar la respuesta. En caso negativo indicar que dependencias se pierden.

La descomposición preserva las dependencias funcionales si la unión de las dependencias que se proyectan en los subesquemas es equivalente al conjunto de dependencias originales.

Sea $K = \{A \rightarrow BC, C \rightarrow DG, BD \rightarrow E\}$

Es necesario determinar si K es equivalente a F .

Por construcción $K^+ \subseteq F^+$. Es necesario determinar si $F^+ \subseteq K^+$.

Las dependencias de F que no pertenecen a K son: $AB \rightarrow D, BC \rightarrow G$.

Se determina si pertenecen a K^+ :

$(AB)^+_K = \{A,B,C,D,G,E\} \Rightarrow AB \rightarrow D \in K^+$.

$(BC)^+_K = \{B,C,D,G,E\} \Rightarrow BC \rightarrow G \in K^+$.

Por lo tanto K es equivalente a F , **ρ preserva las dependencias funcionales.**

5. Sea $J = F \cup \{B \twoheadrightarrow A \mid C\}$. Determinar en que forma normal se encuentra ρ con respecto a J . Justificar la respuesta.

Considerando el conjunto de dependencias J , el único cambio con respecto a F es que en R_1 se cumple la dependencia $B \twoheadrightarrow A$.

R_2 y R_3 cumplen con BCNF como se vio anteriormente, y al no cumplirse dependencias multivaluadas no triviales también cumplen 4NF.

En R_1 , B no es superclave, por lo tanto $B \twoheadrightarrow A$ viola 4NF.

Por lo tanto R_1 se mantiene en BCNF y la descomposición también.

Ejercicio 3 (10 puntos)

La siguiente realidad corresponde a una Clínica Veterinaria en la cual se registra la atención de todas las mascotas que son atendidas. Las tablas más importantes de la base son las siguientes:

Mascotas (matrícula, tipo, raza, nombre, teléfonoDueño): contiene la información de los animales que se atienden en la clínica. La matrícula los identifica, el tipo de animal indica si es perro, gato, pájaro, conejo, etc.; además interesa la raza del mismo (perro bulldog, perro labrador, gato siamés, conejos enanos, etc.), el nombre de la mascota y el teléfono de su dueño.

Veterinarios (ciVet, nombre, teléfono, especialidad): contiene la información de todos los veterinarios de la clínica. La especialidad indica si es cirujano, si trabaja en medicina general o si es anestesista.

Atencion (ciVet, matrícula, fechaAtención): contiene la información de cada una de las atenciones a un animal.

Sea la siguiente consulta SQL:

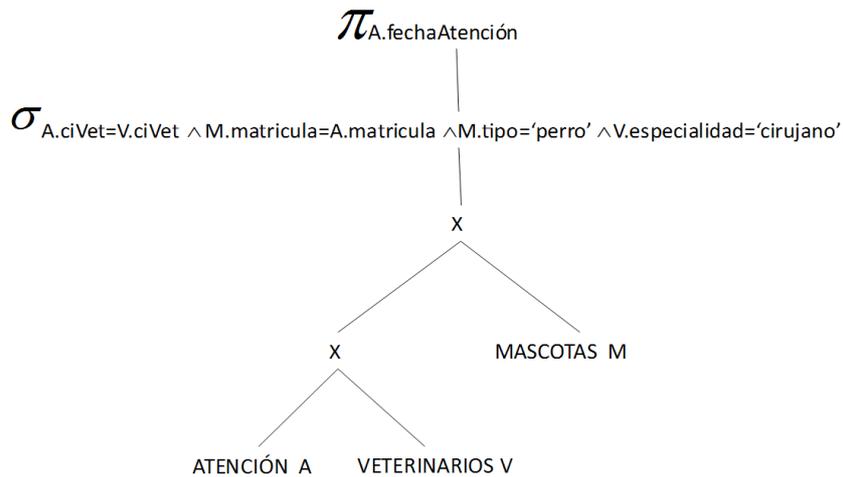
```
SELECT A.fechaAtencion
FROM Atencion A, Veterinarios V, Mascotas M
WHERE A.ciVet = V.ciVet and A.matricula = M.matricula
      and M.tipo = "perro" and V.especialidad = "cirujano";
```

Se dispone de 5 buffers de memoria y se sabe que todos los atributos tienen distribución uniforme (recuerde que $V(A,T)$ es la cantidad de valores distintos que tiene un atributo A en una tabla T). Además, de cada tabla se conoce la siguiente información:

Relación R	n _R	Atributos	Índices
Veterinarios V	100	V(nombre, V) = 25 V(especialidad, V) = 3	- Índice primario sobre ciVet - Índice Sec. árbol b+ sobre especialidad
Mascotas M	4000	V(tipo, M) = 8	- Índice primario sobre matrícula - Índice Sec. árbol b+ sobre tipo - Índice Sec. árbol b+ sobre raza
Atención A	100000		- Índice Primario sobre ciVet, matrícula, fechaAtención

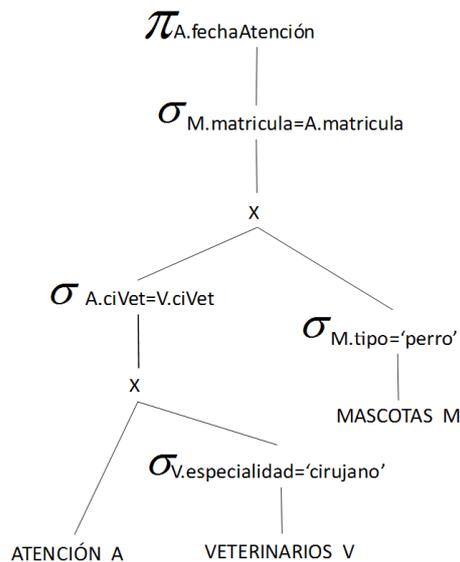
Se pide:

1. El árbol canónico de la consulta dada.
Como se observa, se proyectan todos los atributos que aparecen en el "select" de la consulta, y se seleccionan sólo aquellas tuplas que cumplen la condición del "where". Además, el orden de las tablas debe ser el que aparece en la consulta sql.



2. Aplicar las heurísticas que le permitan llegar al plan lógico optimizado. Explicar cada paso.

Se cambian las selecciones conjuntivas por una cascada de selecciones simples, moviendo dichas selecciones lo más abajo posible en el árbol.

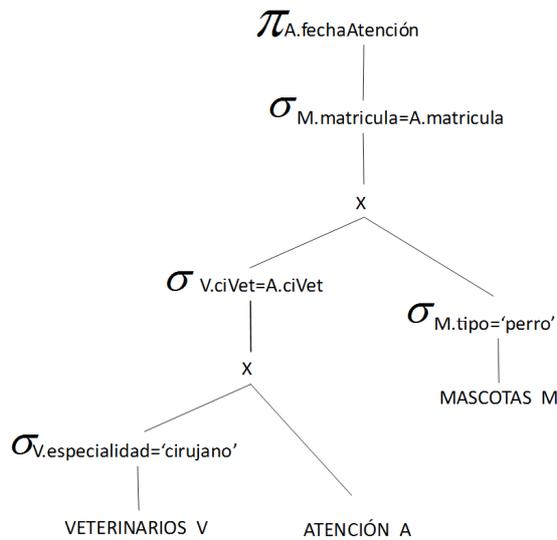


Se colocan a la izquierda de los productos las hojas que generan menos tuplas, asegurando que el orden de las hojas no cause operaciones de producto cartesiano.

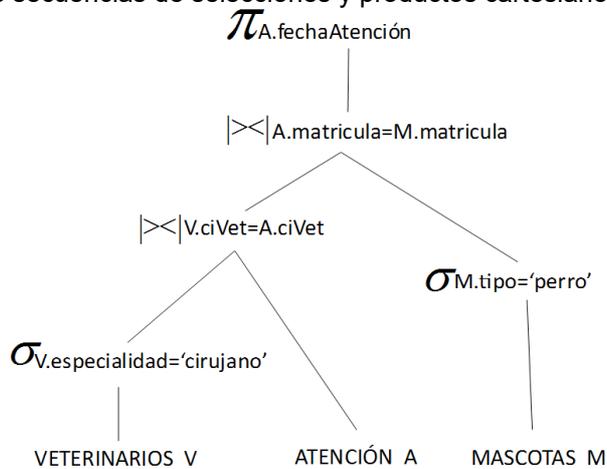
Tabla *Veterinarios*: 100 tuplas
 $V(\text{especialidad}, V) = 3$
 $100/3 = 33$ tuplas

Tabla *Mascotas*: 4000 tuplas
 $V(\text{tipo}, M) = 8$
 $4000/8 = 500$ tuplas

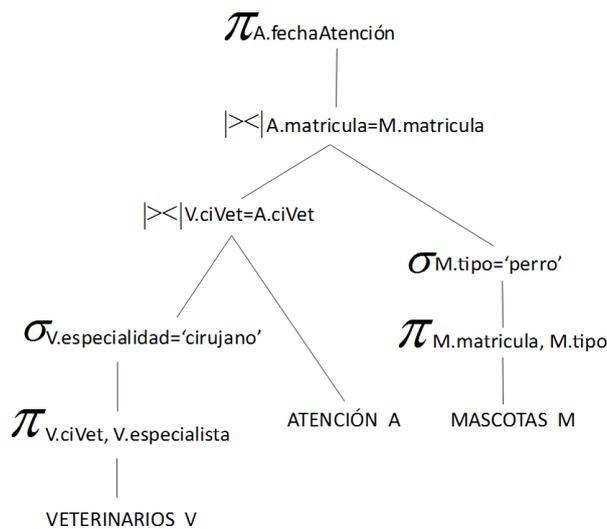
Considerando la heurística, el orden de las hojas en el árbol debe ser *Veterinarios* (33 tuplas), *Mascotas* (500 tuplas) y *Atención* (100000 tuplas). Sin embargo, como las tablas *Veterinarios* y *Mascotas* generan un producto cartesiano no es posible dejarlas en ese orden. Por lo tanto, el orden de las hojas del árbol debe ser *Veterinarios*, *Atención* y *Mascotas*. Esto así porque el plan lógico siempre debe quedar recostado a la izquierda y sólo deben intercambiarse las hojas del árbol. Finalmente, se obtiene el siguiente árbol:



Luego, se deben cambiar las secuencias de selecciones y productos cartesianos por joins.



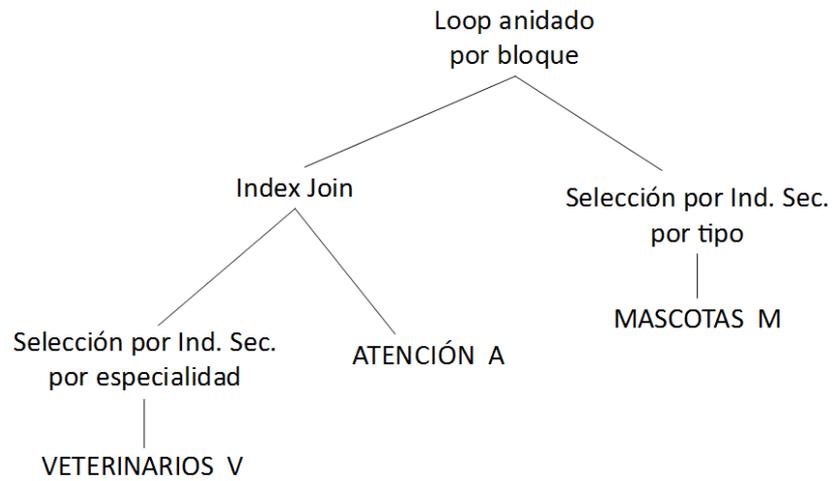
Finalmente, se mueven las proyecciones lo más abajo posible en el árbol, agregando las proyecciones que sean necesarias. El plan lógico que se obtiene a partir del árbol canónico, y aplicando todas las heurísticas, es el que se presenta a continuación:



3. Dar un plan físico para el plan lógico propuesto en la parte anterior, que utilice índices cuando sea posible. Justifique las decisiones tomadas.

Para las selecciones considero los índices secundarios árboles B+, por lo que aplico una selección por índice secundario en ambos casos. En la tala *Atención* tengo un índice primario sobre *ciVet*, *matricula*,

fechaAtención, por lo que es posible utilizar el operador *index join*, esto es así porque *ciVet* es el primer campo del índice de la tabla *Atención*. De lo contrario, si el índice de dicha tabla fuera *matricula*, *ciVet*, *fechaAtención*, esto no sería posible. Finalmente, como ya no tengo más índices del lado derecho (se han perdido al aplicar las operaciones anteriores), considero el *loop anidado por bloque*. El plan físico que se obtiene como resultado es el siguiente:



Ejercicio 4 (15 puntos)

Considere el siguiente programa que trabaja sobre una base de datos de ventas y stock:

```
program venta;
begin

START
input (producto, cant-vendida);

ventas-prod := read (tabla-vtas, producto); /* lee de la bd cantidad vendida del producto */
nueva-ventas := ventas-prod + cant-vendida;
write (tabla-ventas, producto, nueva-ventas); /* actualiza en la bd cantidad vendida del producto */

cant-exist := read (tabla-stock, producto); /* lee de la bd cantidad existente del producto */
if (cant-exist < cant-vendida) {
    output ("cantidad existente insuficiente");
    ABORT
} else {
    nueva-cant := cant-exist - cant-vendida;
    write (tabla-stock, producto, nueva-cant); /* actualiza en la bd cantidad exist. del producto */
    COMMIT
    output ("transacción completa");
}
end program.
```

Representamos esta transacción como:

T: r(X) w(X) r(Y) w(Y) c

Si ocurriera el ABORT, sería

T: r(X) w(X) r(Y) a

PARTE 1

Sean T1 y T2 ejecuciones de T. Dadas las siguientes historias sobre T1 y T2:

H1: r1(X) w1(X) r2(X) w2(X) r2(Y) r1(Y) w1(Y) w2(Y) c1 c2

H2: r1(X) w1(X) r2(X) w2(X) r2(Y) w2(Y) c2 r1(Y) w1(Y) c1

H3: r1(X) w1(X) r1(Y) w1(Y) c1 r2(X) w2(X) r2(Y) w2(Y) c2

a) Contestar para cada una, justificando:

- ¿es recuperable?
- ¿evita abortos en cascada?

b) Considere que para el producto p1 la existencia en stock es de 50, y se ejecutan T1(p1, 60) y T2(p1, 20). Si estas ejecuciones se intentan hacer según las historias H1, H2 y H3, contestar para cada historia: ¿qué sucedería? ¿La BD quedaría consistente?

Nota: T(p, n) significa que en esa transacción se ingresa en el input el producto p, y la cant-vendida n.

Solución:

a)

H1:

- Es recuperable, ya que T2 lee de T1 y se hace c1 antes que c2.
- No EAC, ya que T2 lee de T1 antes que suceda c1.

H2:

- No es recuperable, ya que T2 lee de T1 y se hace c2 antes que c1.
- No EAC, ya que si no es recuperable no puede cumplir EAC.

H3:

- Es recuperable y EAC por ser una historia serial.

b)

H1:

- Si se ejecutan T1(p1,60) y T2(p1,20), sucederán abortos en cascada, ya que T1 abortará antes que T2 haga commit. Entonces el sistema abortará también a T2. La BD quedará consistente.

H2:

- Si se ejecutan T1(p1,60) y T2(p1,20) la BD quedará inconsistente, ya que a1 sucederá después que c2 y por lo tanto T2 no se revertirá.

Sucedirá de la siguiente forma:

$r1(X) w1(X) r2(X) w2(X) r2(Y) w2(Y) c2 r1(Y) a1$

H3:

- Si se ejecutan T1(p1,60) y T2(p1,20), T1 abortará antes que T2 lea el valor de X, por lo tanto T2 va a leer un valor correcto. La BD quedará consistente.

PARTE 2

a) Agregar a T bloqueos y desbloqueos de lectura y escritura tal que se cumpla el protocolo 2PL básico.

T: $rl(X) r(X) wl(X) w(X) rl(Y) r(Y) wl(Y) u(X) w(Y) u(Y) c$

b) Agregar a T bloqueos y desbloqueos de lectura y escritura tal que se cumpla el protocolo 2PL estricto.

T: $rl(X) r(X) wl(X) w(X) rl(Y) r(Y) wl(Y) w(Y) c u(X) u(Y)$

c) Dar una historia de T1 y T2 que contenga los bloqueos y desbloqueos como fueron puestos en la parte a) y decir si es serializable, justificando.

H: $r1(X) r1(X) w1(X) w1(X) r1(Y) r1(Y) w1(Y) u1(X) r2(X) r2(X) w2(X) w2(X) w1(Y) u1(Y) r2(Y) r2(Y) w2(Y) u2(X) w2(Y) u2(Y) c2 c1$

Es serializable porque T1 y T2 siguen el protocolo 2PL.

d) Considere la transacción con bloqueos y desbloqueos propuesta en la parte b). Cuáles de las historias de la Parte 1 (H1, H2 y H3) podrían suceder? Justifique la respuesta.

En la parte b) T1 y T2 siguen el protocolo 2PL estricto. Este protocolo asegura historias estrictas, por lo tanto no pueden haber historias no estrictas cuyas transacciones sigan 2PL estricto.

Recordamos que el conjunto de las historias estrictas está incluido en el conjunto de las historias que cumplen EAC, y el de las historias que cumplen EAC está incluido en el de las historias recuperables.

H1 y H2 no son estrictas porque no cumplen EAC, por lo tanto H1 y H2 no pueden suceder.

H3 es estricta porque es serial. Además los locks no tendrían conflicto con una historia serial. Por lo tanto H3 sí puede suceder.