

## 3er Parcial -1er Cuatrimestre 2017 - Base de Datos

- Cada ejercicio debe realizarse en hojas separadas y numeradas. Debe identificarse **cada hoja** con **nombre, apellido y LU**. Poner el **Nº de Orden** en la primer hoja de CADA EJERCICIO
- Cada tema tiene un criterio de aprobación propio y se recupera por tema. **NO DEBEN COMETERSE ERRORES CONCEPTUALES GRAVES**.
- La devolución de los exámenes corregidos es personal. Los pedidos de revisión se realizarán por escrito, antes de retirar el examen corregido del aula.
- La interpretación del enunciado forma parte de la evaluación.

### 1 Optimización

**Criterio de aprobación:** se aprueba con 6.5 pts, sin errores conceptuales, y **el plan de ejecución propuesto debe ser muy cercano al plan de menor costo posible**.

Se tiene una Base de Datos con el siguiente esquema:

Estudio (idEstudio, nombreEstudio, pais)

Pelicula (idPelicula, nombre, año, genero, idEstudio, idDirector)

Director (idDirector, nombreApellido, nacionalidad)

Los Estudios producen muchas películas. Las películas son dirigidas por un director y producidas por un estudio. Los directores pueden dirigir muchas películas

Se desea optimizar la siguiente consulta:

```
SELECT D.nombre, P.Nombre, P.Año,
FROM Estudio E , Pelicula P, Director D
WHERE E.idEstudio = P.idEstudio AND
      P.idDirector = D.idDirector AND
      P.año >= 2010 AND P.año <= 2015 AND
      P.Genero = "Comedia" AND
      E.Pais = "USA"
```

Se tienen la siguiente información sobre índices y tamaños:

- Índices ( altura de árboles B+ es  $X = 3$ ):
    - $I_1$  índice B+ unclustered sobre  $D.nombreApellido$
    - $I_2$  índice B+ unclustered sobre  $D.idDirector$
    - $I_3$  índice B+ unclustered sobre  $P.pais$
    - $I_4$  índice B+ unclustered sobre  $P.Genero$
    - $I_5$  índice hash sobre  $P.año$  con máximo 10 bloques por bucket.
  - No hay dos directores con el mismo nombre.
  - Hay 100.000 películas de comedia.
  - Se producen 2.500 películas por año.
  - $I_{Estudio.pais} = 10$ ,  $T_{Pelicula} = 350.000$ ,  $T_{Director} = 100.000$ ,  $T_{Estudios} = 3.000$
  - Tamaño de los bloques:  $LB = 4096$  bytes. Tamaño de todos los atributos 32 bytes.
  - La cantidad de bloques disponibles de memoria es  $B = 204$ .
  - Suponga distribución uniforme. Asumir que los punteros a tupla que se necesiten recorrer en los índices B+ entran en una hoja.
- a) Escriba el árbol canónico y partir de él modifíquelo para obtener un plan de ejecución optimizado indicando solamente el árbol final. Justifique cada una de las decisiones que tome para su propuesta de plan de ejecución. Justifique también porque decide utilizar o no utilizar cada uno de los índices disponibles.
- b) Calcule el costo del árbol optimizado.

## 2 Concurrencia y recuperabilidad

**Criterio de aprobación:** se aprueba con 6.5 pts. Valores parciales: a) 4, b) 3, c) 3. Debe tener **al menos algo correcto en cada ítem**

a) Dadas las transacciones  $T_1, T_2, T_3, T_4$  y  $T_5$  y la siguiente historia:

$H = w_4(X); w_1(Y); r_3(Z); w_2(W); r_2(Y); w_2(V); w_1(S); r_3(W); w_3(Y); r_1(Z); w_4(S); r_4(Z); c_4;$   
 $r_3(S); c_1; w_3(X); c_3; c_2; w_5(W); r_5(X); c_5$  Se pide

- Construir el SG(H) e indicar si H es SR. En caso afirmativo indicar las historias seriales equivalentes.
- Clasificar H con respecto a los criterios de Recuperabilidad.
- Reescribir otra historia H1 basada en las mismas  $T_1, T_2, T_3, T_4$  y  $T_5$  tal que sea RC pero no ACA. Justificar la respuesta.
- Reescribir  $T_1, T_2, T_3, T_4$  y  $T_5$  en el modelo de **Locking Ternario** tal que respeten el protocolo 2PL.

**Justifique todas sus respuestas**

b) Dada la historia de a) otorgue *timestamps* y ubique los *st* de cada transacción, utilizando un protocolo de TimeStamp con Multiversion de modo tal que no se aborte ninguna transacción

c) Suponga que el estado del log después de un crash es el detallado a continuación (el crash se produce luego de la línea 10):

- < START  $T_1$  >
- <  $T_1, X, 3, 9$  >
- < START  $T_2$  >
- <  $T_2, Y, 8, 10$  >
- < START  $T_3$  >
- <  $T_3, W, 13, 7$  >
- < START CKPT ( $T_1, T_2, T_3$ ) >
- < START  $T_4$  >
- <  $T_4, Z, 11, 22$  >
- < COMMIT  $T_4$  >

Asumiendo un mecanismo de recuperación Undo/Redo con checkpoints no quiescente especifique claramente, justificando las respuestas:

- Las transacciones a deshacer.
- Los cambios efectuados en la base de datos, y el orden de los mismos
- Los cambios efectuados en el log.