

Segundo Parcial de Fundamentos de Bases de Datos

Noviembre 2023

Soluciones

Duración: 3 horas

Presentar la resolución del parcial:

- Con las hojas numeradas y escritas de un solo lado. Comience cada ejercicio en una hoja nueva.
- 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 (18 puntos)

Se desea construir una base de datos para almacenar información sobre cines, películas, funciones y espectadores.

Los atributos que se definen para esta base de datos son los siguientes:

Atributo	Descripción
nomC	Nombre del cine
dir	Dirección del cine
cap	Capacidad total del cine
nomS	Nombre de sala
calidad	Calidad de la sala
titulo	Título de la película
dur	Duración de la película
gen	Género de la película
fecha	Fecha de la función
hini	Hora inicio de la función
precio	Precio de la función
ci	Cédula de la persona
nomP	Nombre de la persona
snack	Si una persona compra o no snack para la función

Las dependencias funcionales que se cumplen entre ellos son las siguientes:

$F = \{ \text{nomC} \rightarrow \text{dir}, \text{cap}$
 $\text{nomC}, \text{nomS} \rightarrow \text{calidad}$
 $\text{titulo} \rightarrow \text{dur}$
 $\text{fecha}, \text{hini}, \text{titulo}, \text{nomS}, \text{nomC} \rightarrow \text{precio}$
 $\text{ci} \rightarrow \text{nomP}$
 $\text{ci}, \text{fecha}, \text{hini}, \text{titulo}, \text{nomS}, \text{nomC} \rightarrow \text{snack} \}$

Se pide resolver:

a) Suponiendo que alguien descompuso la relación universal R(nomC, dir, cap, nomS, calidad, titulo, dur, gen, fecha, hini, precio, ci, nomP, snack) en las siguientes dos relaciones:

R1 (nomC, dir, cap, nomS, calidad, titulo, dur, gen)

R2 (fecha, hini, titulo, nomS, nomC, precio, ci, nomP, snack)

Decir si esta descomposición tiene Join sin Pérdida.

b) Decir si el siguiente conjunto de dependencias funcionales es un cubrimiento minimal de F, justificando.

G = { nomC → dir
 nomC → cap
 nomC, nomS → calidad
 titulo → dur
 fecha, hini, titulo, nomS, nomC → precio
 ci → nomP
 ci, fecha, hini, titulo, nomS, nomC → snack }

c) Buscar todas las claves de la relación universal R según F, justificando.

d) Partiendo de la relación universal R dar una descomposición en 3NF con JSP y preservación de dependencias.

e) Llevar la descomposición de la parte anterior a una descomposición en BCNF.

Solución:

a)

Para ver si la descomposición tiene JSP utilizamos la siguiente propiedad:

D = (R1, R2) de R tiene JSP respecto a F sobre R sii:

- $R_1 \cap R_2 \rightarrow R_1 - R_2 \in F^+$

o
 - $R_1 \cap R_2 \rightarrow R_2 - R_1 \in F^+$

Calculamos $R_1 \cap R_2, R_1 - R_2$ y $R_2 - R_1$:

$R_1 \cap R_2 = \{ nomC, nomS, titulo \}$

$R_1 - R_2 = \{ dir, cap, calidad, titulo, dur, gen \}$

$R_2 - R_1 = \{ fecha, hini, precio, ci, nomP, snack \}$

$(R_1 \cap R_2)^+ = \{ nomC, nomS, titulo, dir, cap, calidad, dur \}$

Se puede observar que $R_1 - R_2 \not\subseteq (R_1 \cap R_2)^+$ y $R_2 - R_1 \not\subseteq (R_1 \cap R_2)^+$, por lo que no se cumple ninguna de las dependencias ($R_1 \cap R_2 \rightarrow R_1 - R_2$ o $R_1 \cap R_2 \rightarrow R_2 - R_1$).

Luego, la descomposición **no** tiene JSP.

b)

Primero vamos a ver que G cubre a F.

Aplicando la regla RI5 (unión) a las dfs nomC → dir y nomC → cap, obtengo la df nomC → dir, cap. Luego el nuevo conjunto G' con la nueva df deducida cumple que G' = F. Luego, G cubre a F.

Ahora vemos que G es un cubrimiento minimal de F.

1. No hay df con más de un atributo a la derecha.
2. Para aquellas df que tienen más de un atributo a la izquierda, vemos si hay atributos redundantes

$nomC, nomS \rightarrow calidad$:

- $(nomS)^+ = \{nomS\}$ y $calidad \notin (nomS)^+$
- $(nomC)^+ = \{nomC, dir, cap\}$ y $calidad \notin (nomC)^+$

$fecha, hini, titulo, nomS, nomC \rightarrow precio$:

- Como el atributo “precio” no aparece del lado derecho de ninguna otra dfs, no se puede deducir a partir de los atributos “hini, titulo, nomS, nomC”. A su vez, el atributo “fecha” tampoco aparece del lado derecho de las dfs, por lo que tampoco es posible deducirlo a partir de los atributos “hini, titulo, nomS, nomC”.
- El mismo análisis se puede hacer para los atributos hini, titulo, nomS y nomC. Luego, no hay atributos redundantes en la dependencia.

$ci, fecha, hini, titulo, nomS, nomC \rightarrow snack$

- Como el atributo “snack” no aparece del lado derecho de ninguna otra dfs, no se puede deducir a partir de los atributos “fecha, hini, titulo, nomS, nomC”. A su vez, el atributo “ci” tampoco aparece del lado derecho de las dfs, por lo que tampoco es posible deducirlo a partir de los atributos “fecha, hini, titulo, nomS, nomC”.
- El mismo análisis se puede hacer para los atributos fecha, hini, titulo, nomS y nomC. Luego, no hay atributos redundantes en la dependencia.

3. Como ningún atributo aparece a la derecha en más de una df, no hay dfs redundantes.

Luego, G es un cubrimiento minimal.

c)

ATRIBUTOS_SOLO_A_LA_IZQUIERDA = { nomC, nomS, titulo, fecha, hini, ci }

ATRIBUTOS_SOLO_A_LA_DERECHA = { dir, cap, calidad, dur, precio, nomP, snack }

ATRIBUTOS_EN_NINGUNA_DF = { gen }

ATRIBUTOS_A_AMBOS_LADOS = { }

$(nomC, nomS, titulo, fecha, hini, ci, gen)^+ = \{nomC, nomS, titulo, fecha, hini, ci, gen, dir, cap, calidad, dur, precio, nomP, snack\} = R$

Luego, $(nomC, nomS, titulo, fecha, hini, ci, gen)$ es la única clave; ya que contiene los atributos que tienen que estar en todas las claves (no podemos sacar ningún atributo), por lo que cualquier atributo que agreguemos va a generar una superclave.

d)

Aplicamos el algoritmo de Descomposición en 3NF con JSP y pres de dfs, partiendo del conjunto G; ya que este es un cubrimiento minimal de F.

$R_1(nomC, dir, cap)$

$R_2(nomC, nomS, calidad)$

$R_3(titulo, dur)$

$R_4(fecha, hini, titulo, nomS, nomC, precio)$

$R_5(ci, nomP)$

$R_6(ci, fecha, hini, titulo, nomS, nomC, snack)$

Agregamos un er con la clave de R.

$R_7(nomC, nomS, titulo, fecha, hini, ci, gen)$

Como no hay esquemas redundantes, no se elimina ningún er.

e)

$R_1(\text{nomC}, \text{dir}, \text{cap}) \quad \Pi_{R_1}(F) = \{ \text{nomC} \rightarrow \text{dir}, \text{cap} \}$

$R_2(\text{nomC}, \text{nomS}, \text{calidad}) \quad \Pi_{R_2}(F) = \{ \text{nomC}, \text{nomS} \rightarrow \text{calidad} \}$

$R_3(\text{titulo}, \text{dur}) \quad \Pi_{R_3}(F) = \{ \text{titulo} \rightarrow \text{dur} \}$

$R_4(\text{fecha}, \text{hini}, \text{titulo}, \text{nomS}, \text{nomC}, \text{precio}) \quad \Pi_{R_4}(F) = \{ \text{fecha}, \text{hini}, \text{titulo}, \text{nomS}, \text{nomC} \rightarrow \text{precio} \}$

$R_5(\text{ci}, \text{nomP}) \quad \Pi_{R_5}(F) = \{ \text{ci} \rightarrow \text{nomP} \}$

$R_6(\text{ci}, \text{fecha}, \text{hini}, \text{titulo}, \text{nomS}, \text{nomC}, \text{snack})$

$\Pi_{R_6}(F) = \{ \text{ci}, \text{fecha}, \text{hini}, \text{titulo}, \text{nomC}, \text{nomS} \rightarrow \text{snack} \}$

$R_7(\text{nomC}, \text{nomS}, \text{titulo}, \text{fecha}, \text{hini}, \text{ci}, \text{gen}) \quad \Pi_{R_7}(F) = \{ \}$

En todos los er las dfs que se proyectan cumplen tener una superclave a la izquierda. Por lo tanto, en ningún er hay dfs que violen BCNF.

Luego, la descomposición está en BCNF.

Ejercicio 2 (12 puntos)

Se desea modelar una base de datos con información sobre parques de diversiones, muy similar a la del ejercicio de modelado del primer parcial 2023.

Cada parque de diversiones tiene un código que lo identifica (**codP**), un nombre (**nomP**) y una dirección (**dirP**). Los parques de diversiones contienen distintas atracciones, las cuales poseen un código que las identifica dentro del parque (**codA**). Además, tienen un nombre (**nomA**) y tipo (montaña rusa, infantil, etc.) (**tipA**).

En las atracciones trabajan personas, pudiendo una persona trabajar en varias atracciones. De las personas se conoce la cédula (**ciPe**), el nombre (**nomPe**) y la edad (**edadPe**).

La entrada a los parques es gratuita, pero para acceder a una atracción es necesario comprar un boleto. Los boletos individuales tienen un número que los identifica (**numBI**), un precio (**precioBI**), los compra una persona y valen para una atracción, y tienen una hora y período de validez (Ejemplo: 15hs. – 2 horas) (**valBI**).

Por otro lado, existen grupos de personas, los cuales se identifican por su nombre (**nomG**) y se registran las personas que integran cada grupo. Una misma persona puede integrar varios grupos. Los parques disponen de boletos grupales, que tienen un número que los identifica (**numBG**), un precio (**precioBG**), que sólo pueden ser comprados por los grupos. De estos boletos se conoce la fecha de compra (**fechaB**) y son válidos para un conjunto de atracciones dentro de un mismo parque.

Por último, los parques organizan distintos eventos (Ejemplo: conciertos, festivales, desfiles, etc.) de los cuales se conoce un identificador (**codE**), el nombre (**nomE**) y la descripción (**descE**). Un mismo evento se puede repetir en el tiempo y realizarse en varios parques (Ejemplo: El desfile de Navidad se hace todos los años en el Parque 1, Parque 2 y Parque 3), siendo cada una de esas una edición diferente. De cada edición del evento, se almacena la fecha (**fechaEd**), hora (**horaEd**) y parque en el cual se realiza. En un mismo parque no se realiza más de un evento en simultáneo.

Se pide:

a) Indicar qué dependencias funcionales se cumplen en la realidad

b) Considerando que existe la siguiente tabla en el esquema relacional:

Personas (ciPe, codA, codP, nomG)

Almacena la información de en qué atracciones trabajan las personas y a qué grupos pertenecen.

Si se cumple alguna dependencia multivaluada, expresarla y explicar por qué considera que se cumple (puede utilizar un pequeño ejemplo para mostrarlo).

Solución:

a)

$\text{codP} \rightarrow \text{nomP}, \text{dirP}$

$\text{codA}, \text{codP} \rightarrow \text{nomA}, \text{tipA}$

$\text{ciPe} \rightarrow \text{nomPe}, \text{edadPe}$

$\text{numBI} \rightarrow \text{precioBI}, \text{valBI}, \text{codA}, \text{codP}, \text{ciPe}$

$\text{numBG} \rightarrow \text{codP}, \text{precioBG}, \text{nomG}, \text{fechaB}$

$\text{codE} \rightarrow \text{nomE}, \text{descE}$

$\text{fechaEd}, \text{horaEd}, \text{codP} \rightarrow \text{codE}$

b)

De la descripción de la realidad se sabe que una persona puede trabajar en más de una atracción y una persona puede pertenecer a más de un grupo, independientemente de las atracciones en las cuales trabaja por lo que se cumple la dmV:

$\text{ciPe} \twoheadrightarrow \text{codA}, \text{codP}$

y por complemento, se cumple $\text{ciPe} \twoheadrightarrow \text{nomG}$.

Ejercicio 3 (15 puntos)

Considere el siguiente esquema relacional de asignación de viajes de una agencia de transporte nacional.

CARRETERAS (ciuOrigen, ciuDestino, cantKm, tipo)

Contiene información sobre las carreteras que brinda la agencia. El tipo de carretera puede ser 'camino', 'avenida', 'ruta', 'autopista'.

EMPLEADOS (ciEmp, nombre, tel, dir)

Datos de los empleados de la empresa.

VIAJES (ciuOrigen, ciuDestino, fecha, hora, ciEmp)

Datos de los viajes de la agencia, es decir en qué fecha y hora se realiza cada ruta y los empleados que están involucrados en la misma.

$$\begin{aligned}\Pi_{\text{ciuOrigen}, \text{ciuDestino}}(\text{VIAJES}) &\subseteq \Pi_{\text{ciuOrigen}, \text{ciuDestino}}(\text{CARRETERAS}) \\ \Pi_{\text{ciEmpleado}}(\text{VIAJES}) &\subseteq \Pi_{\text{ciEmpleado}}(\text{EMPLEADOS})\end{aligned}$$

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	Factor bloqueo	Atributos	Índices
CARRETERAS C	2000	10	V(tipo, C) = 4	Índice primario por atributos cuiOrigen, cuiDestino, IndCiudades Índice secundario por atributo tipo, IndTipo
EMPLEADOS E	80	20		Índice primario por ciEmp, IndCi
VIAJES V	10000	10	El 40% de los viajes se realizan por rutas.	Índice secundario por ciEmp, IndViajesCi

Considere la siguiente consulta sobre el esquema dado:

```

SELECT cuiOrigen, nombre, cantKm
FROM VIAJES V, EMPLEADOS E, CARRETERAS C
WHERE V.ciEmp = E.ciEmp AND V.ciuOrigen = C.ciuOrigen
        AND V.ciuDestino = C.ciuDestino AND C.tipo = 'ruta';

```

Se pide:

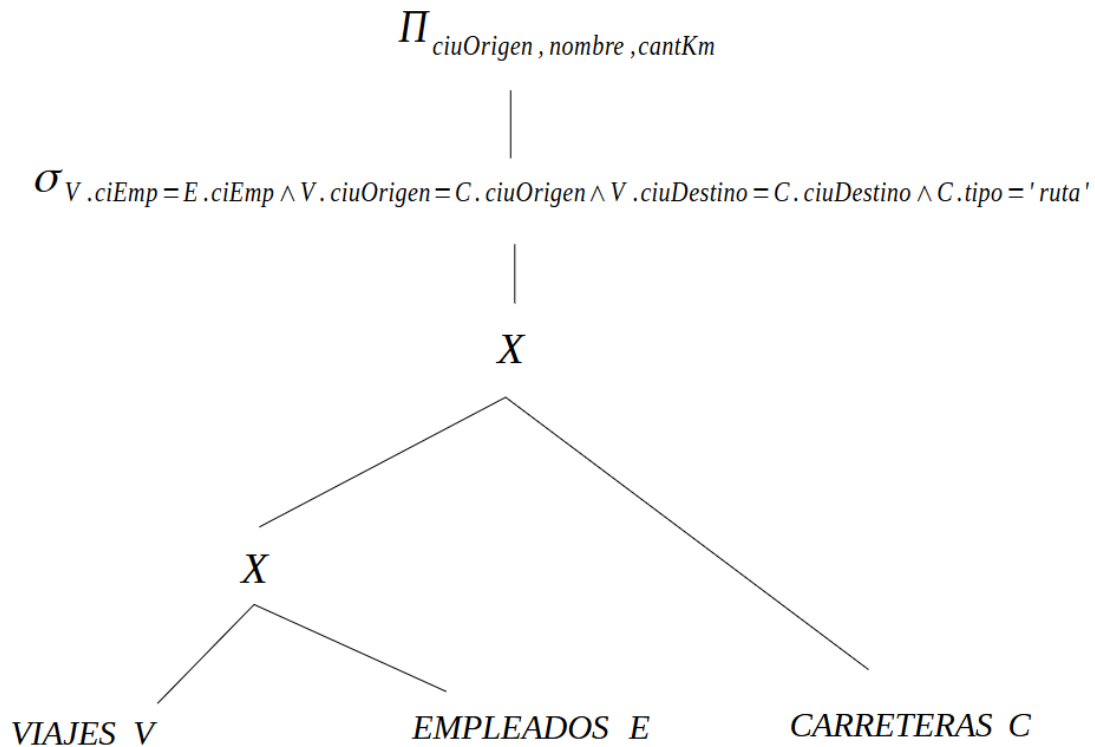
1. Dar el árbol canónico para la consulta.
2. Aplicar las heurísticas para optimización llegando al plan lógico optimizado. Explique cada uno de los pasos ejecutados.
3. Calcular todos los tamaños intermedios y el tamaño final de la consulta.
4. Dar un plan físico para el plan lógico obtenido en la **parte 3**, utilizando índices en los casos en que sea posible.
5. Suponiendo que el índice IndTipo es un índice de 3 niveles (x=3), calcular el costo de la selección que se realiza sobre la tabla CARRETERAS, primero utilizando el índice, y luego sin utilizar el índice, es decir, con una búsqueda lineal.

Nota: Las siguientes son las fórmulas de cálculo de costo para los algoritmos requeridos de selección.

- Selección utilizando índice secundario: $x + s$
- Selección por búsqueda lineal: b_R

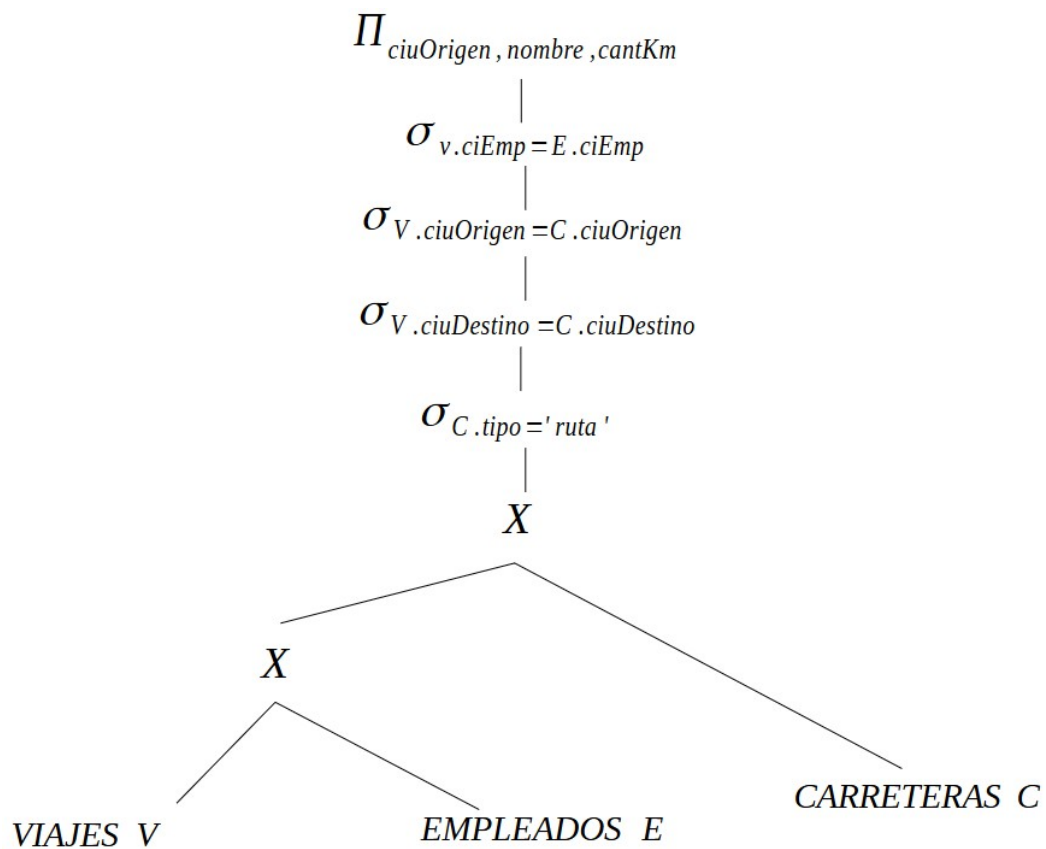
Solución

1.

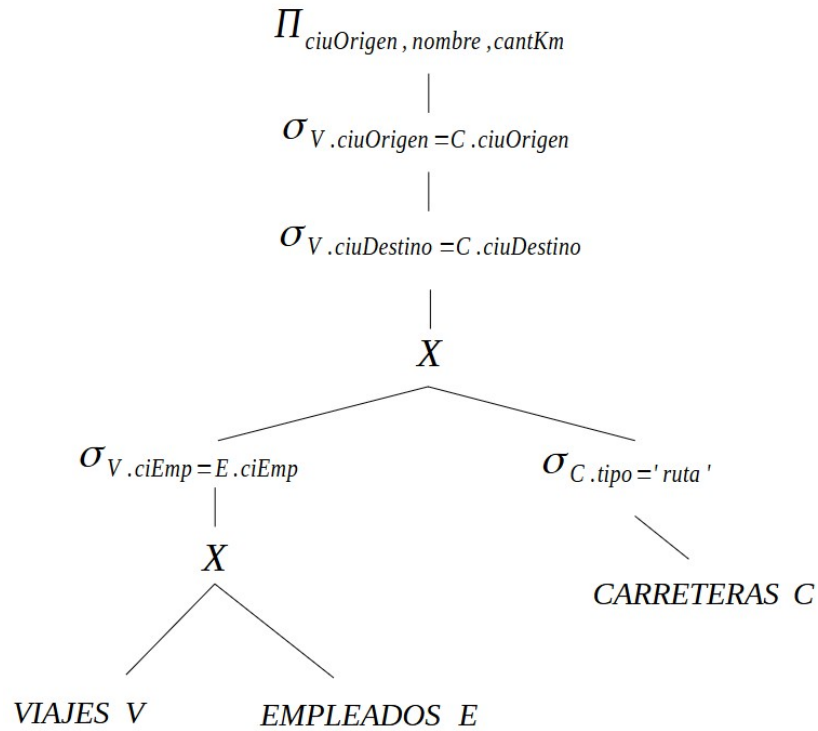


2.

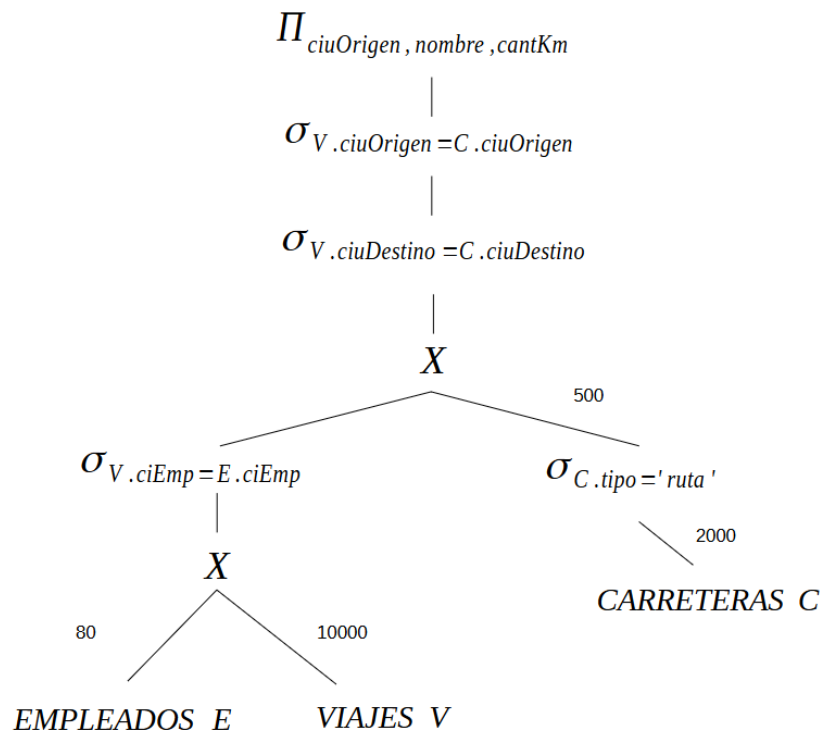
Paso 1: separar las selecciones en cascada de selecciones



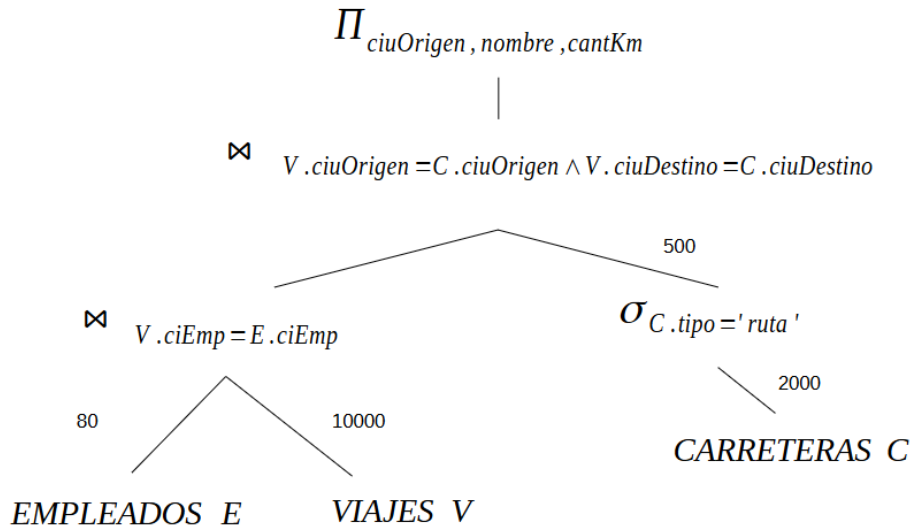
Paso 2: Llevar las selecciones lo más abajo posible



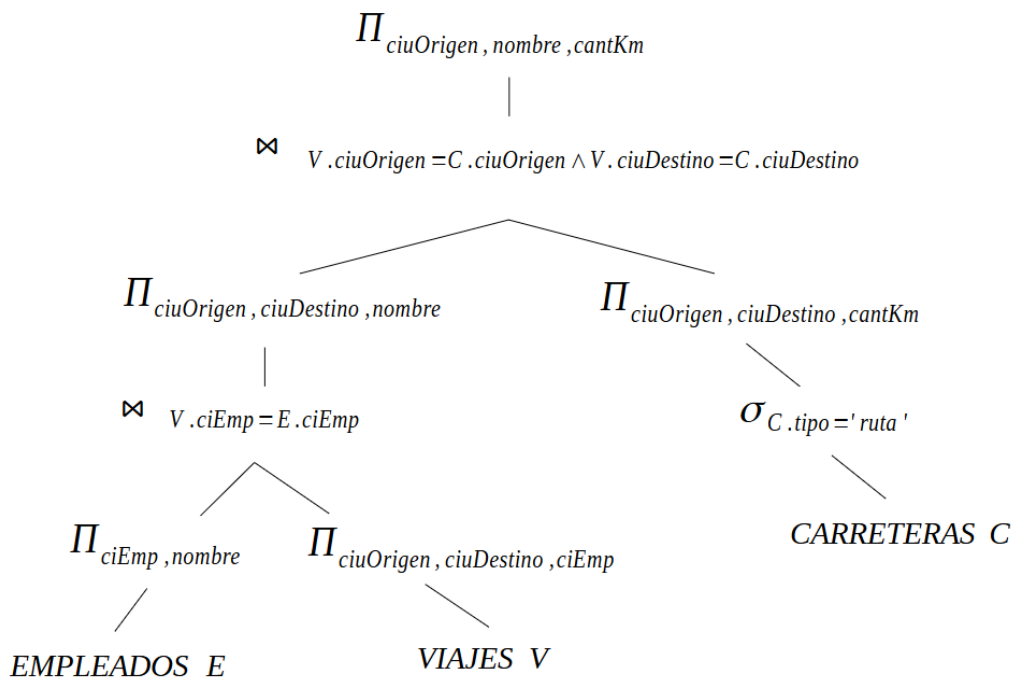
Paso 3: Mover a la izquierda las hojas con menos tuplas, sin generar productos cartesianos innecesarios



Paso 4: Cambiar productos cartesianos y selecciones por joins



Paso 5: Agregar todas las proyecciones necesarias para manejar la menor cantidad de datos posibles



3.

Join entre EMPLEADOS y VIAJES:

Como $\Pi_{ciEmpleado}(VIAJES) \subseteq \Pi_{ciEmpleado}(EMPLEADOS)$, toda tupla de VIAJES tiene correspondiente en la tabla EMPLEADOS. Luego, el tamaño del join es igual a la cantidad de tuplas en VIAJES (10000).

Selección por tipo de CARRETERAS:

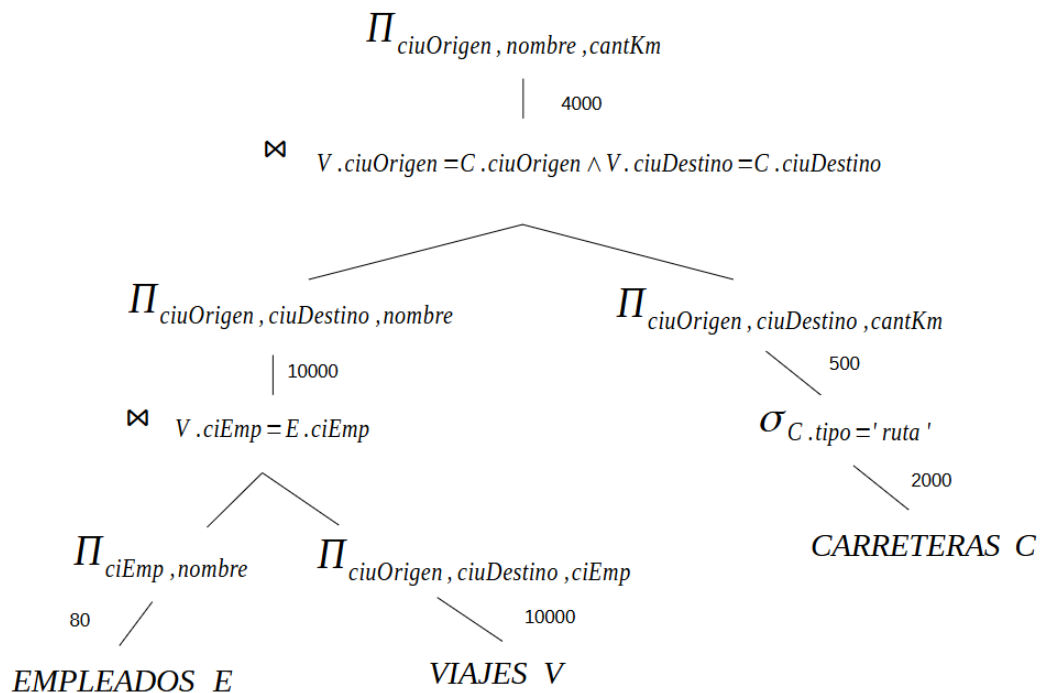
Como se sabe que todos los atributos tienen distribución uniforme y $V(\text{tipo}, C) = 4$, se puede calcular el costo de la selección como $n_C \times \frac{1}{V(\text{tipo}, C)} = 2000 \times \frac{1}{4} = 500$

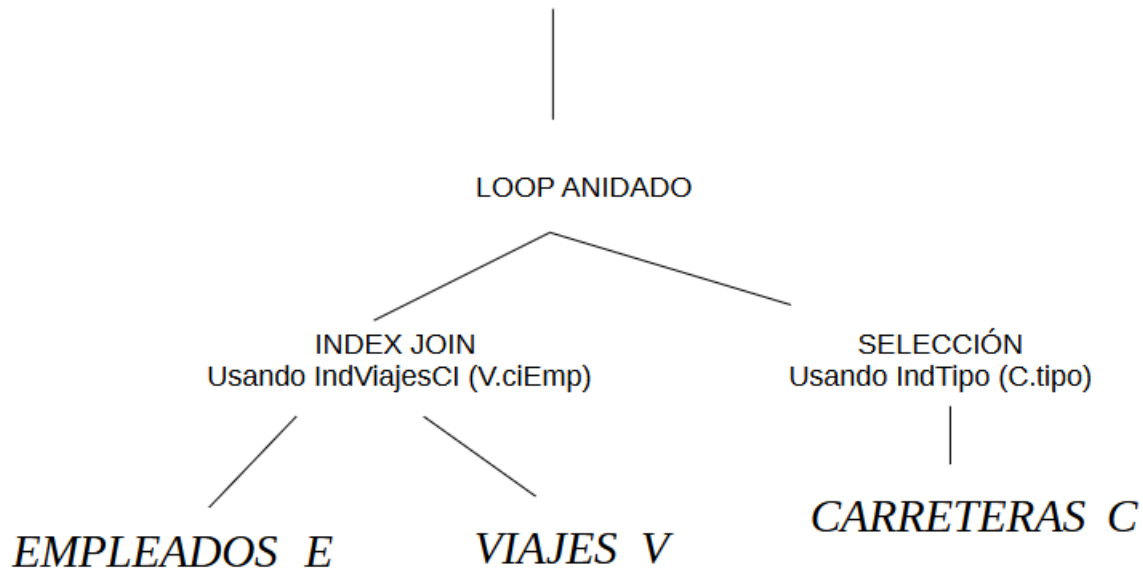
Join con CARRETERAS:

Se sabe que el 40% de los viajes se realizan por rutas. Como del join entre VIAJES y EMPLEADOS obtenemos todas las tuplas de VIAJES, basta con calcular el 40% de 10000 tuplas para obtener el resultado de este join.

100% ----- 10000

40% ----- $x \Rightarrow x = \frac{40 \times 10000}{100} = 4000$





Aclaración: El segundo join no se puede hacer utilizando el índice IndCiudades porque al haber una operación antes (la selección), el join recibe como entrada una tabla intermedia sobre la que ya no tenemos índice.

5.

Costo de selección con índice: $x + s = 3 + 500 = 503$

siendo x la cantidad de niveles del índice y s la cardinalidad de la selección (es decir la cantidad de tuplas del resultado de la selección).

Costo de selección sin índice: $b_R = \frac{\text{Cantidad de registros}}{\text{Cantidad de registros en un bloque}} = \frac{2000}{10} = 200$

siendo b_R la cantidad de bloques de la relación.

Ejercicio 4 (15 puntos)

Considere las siguientes transacciones:

T_1 : $r_1(x)$, $w_1(y)$, $w_1(x)$, $r_1(y)$, c_1

T_2 : $r_2(y)$, $w_2(x)$, $w_2(y)$, c_2

a) Decir, justificando las respuestas, si la historia H es:

- 1) Serializable
- 2) Recuperable
- 3) EAC
- 4) Estricta

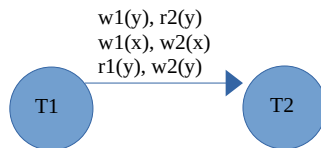
H : $r_1(x)$, $w_1(y)$, $r_2(y)$, $w_1(x)$, $r_1(y)$, c_1 , $w_2(x)$, $w_2(y)$, c_2

- b) Escriba T_1 y T_2 agregándoles locks y unlocks, tal que sigan el protocolo 2PL.
- c) Dar una historia de las T_1 y T_2 dadas en la parte b).
- d) Si es posible. dar una historia de las T_1 y T_2 dadas en la parte b), donde se genere una situación de deadlock.

Solución:

a)

1) Serializable



Es serializable porque el grafo no tiene ciclos.

2) Recuperable

T_2 lee de T_1 (y T_1 no lee de T_2) y T_1 confirma antes que T_2 . Luego, H es recuperable.

3) EAC

No es EAC porque T_2 lee de T_1 pero lo hace antes que T_1 confirme.

4) Estricta

Como no es EAC, tampoco es Estricta; ya que se cumple $Estrictas \subseteq EAC \subseteq Recuperables$

b)

T_1 con locks: $rl_1(x)$, $r_1(x)$, $wl_1(y)$, $w_1(y)$, $wl_1(x)$, $w_1(x)$, $rl_1(y)$, $r_1(y)$, $u_1(x)$, $u_1(y)$, c_1

T_2 con locks: $rl_2(y)$, $r_2(y)$, $wl_2(x)$, $w_2(x)$, $wl_2(y)$, $w_2(y)$, $u_2(y)$, $u_2(x)$, c_2

c)

H : $rl_1(x)$, $r_1(x)$, $wl_1(y)$, $w_1(y)$, $wl_1(x)$, $w_1(x)$, $rl_1(y)$, $rl_2(y)$, $r_2(y)$, $r_1(y)$, $u_1(x)$, $u_1(y)$, c_1 , $wl_2(x)$, $w_2(x)$, $wl_2(y)$, $w_2(y)$, $u_2(y)$, $u_2(x)$, c_2

d)

En la siguiente historia (parcial) de las T_1 y T_2 de la parte (b) se genera un deadlock:

T1	T2
$rl_1(x)$	
$r_1(x)$	
	$rl_2(y)$
	$r_2(y)$

	wl2(x) – T2 espera por T1
wl1(y) – T1 espera por T2	