

IIC 2413 – Bases de Datos
Interrogación 2

Pregunta 1: Procedimientos almacenados

Para esta pregunta considere el siguiente esquema:

```
R(a int, b int, PRIMARY KEY(a, b))  
S(b int, c int, PRIMARY KEY(b, c))
```

Tú necesitas hacer la siguiente consulta sobre el esquema:

```
SELECT *  
FROM R LEFT JOIN S  
ON R.b = S.b;
```

La consulta utiliza el operador `LEFT OUTER JOIN`, pero tu usuario no tiene los permisos para utilizar este comando, entre varios otros. Afortunadamente, tu base de datos soporta el lenguaje PL-PgSQL, por lo que puedes hacer un procedimiento almacenado que compute el resultado de la consulta. En esta pregunta se pide que hagas un procedimiento almacenado llamado `outer_a_mano()` que compute la consulta señalada. Dentro del procedimiento está permitido hacer consultas del tipo:

- `SELECT * FROM R`
- `SELECT * FROM S`
- `CREATE TABLE`
- `DROP TABLE`
- `INSERT INTO`

Hint: Puede insertar nulos en una tabla.

Pregunta 2: Recursión y Programación + SQL

Suponga que usted tiene la siguiente relación:

```
Caminos(  
  ciudad_origen VARCHAR(100),  
  ciudad_destino VARCHAR(100),  
  nombre_camino VARCHAR(100)  
)
```

Que indica los pares de ciudades conectadas entre sí, y el nombre del camino que las conecta. Una instancia de la relación puede ser:

Considere ahora la siguiente consulta:

ciudad_origen	ciudad_destino	nombre_camino
Santiago	Talca	Ruta A
Talca	Temuco	Ruta A
Temuco	Pto Montt	Ruta A
Pto Montt	Hornopirén	Ruta B
Hornopirén	Chaitén	Ruta B
Pto Natales	Pta Arenas	Ruta C

Cuadro 1: Ejemplo de instancia.

```
WITH RECURSIVE Alcanzo(co, cd, n_camino) AS (
  SELECT * FROM Caminos
  UNION
  SELECT C.ciudad_origen, A.cd, A.n_camino
  FROM Caminos C, Alcanzo A
  WHERE C.ciudad_destino = A.co AND C.nombre_camino = A.n_camino
)
SELECT * FROM Alcanzo;
```

a) [1 pto] ¿Qué representa el resultado de la consulta anterior?

Ahora considera una base de datos PSQL en la que no puedes realizar consultas recursivas y de otro tipo por un tema de permisos, pero tienes acceso a conectar la base de datos con tu lenguaje de programación favorito ¹.

b) [5 ptos] Haz una función en Python llamada `calcular_recursion()` que imprima en consola el resultado de la consulta anterior.

Sólo puedes usar consultas del tipo:

- SELECT - FROM - WHERE
- CREATE TABLE
- DROP TABLE
- INSERT INTO

En esta pregunta debes asumir que los datos **no caben en memoria**. Además está prohibido importar librerías adicionales, salvo las necesarias para usar PostgreSQL en tu lenguaje de programación favorito.

c) **Bonus [0.4 ptos]** Indica los nombres, o bien las ciudades que conectan las siguientes rutas en Chile:

- Ruta 5
- Ruta 7
- Ruta 9
- Ruta 78

¹Que esperamos que sea Python.

Pregunta 3: Índices y algoritmos

Índices [2.5 pts]

Considera la relación `Persona(id int PRIMARY KEY, nombre varchar(100), edad int)`. En esta relación la primary key está indexada con un B+Tree *clustered*, mientras que la edad está indexada con un Hash Index *unclustered*. La relación tiene 1 millón de tuplas, y en cada página caben 100 tuplas. El árbol es de altura h y las hojas están ocupadas al 75%. El Hash Index tiene una página por *bucket* (i.e. es suficientemente grande y no tiene *overflow pages*), y cada edad va a parar a un *bucket* distinto.

a) [1.5 ptos] Indique el costo en I/O de las siguientes consultas:

1. `SELECT * FROM PERSONA WHERE id=1234`
2. `SELECT * FROM PERSONA WHERE nombre='Nicholas Castle'`
3. `SELECT * FROM PERSONA WHERE edad=33`
4. `SELECT * FROM PERSONA WHERE id <10000`
5. `SELECT * FROM PERSONA WHERE edad >= 20 and edad <30`

b) [1 pto] Un colega te propone que para agilizar la consulta por edades, deberías armar un índice de tipo *clustered* por las edades. Si bien esto agilizaría la rapidez de las consultas por edad, tú no estás seguro si es buena idea. ¿Qué le responderías a tu colega?.

Algoritmos [3.5 pts]

Considere las relaciones `R(a int, b int)` y `S(b int, c int)`, en donde las relaciones están indexadas por un B+Tree clustered. Tienes acceso a Python y `psycopg2`, pero sólo puedes hacer consultas a la base de datos del tipo:

- `SELECT * FROM R`
- `SELECT * FROM S`
- `CREATE TABLE`
- `DROP TABLE`
- `INSERT INTO`

- a) [2.5 ptos] Entregue un algoritmo en Python que haga uso la librería `psycopg2` que calcule la consulta `SELECT * FROM R, S WHERE R.b = S.b`, pero que lea cada tupla una única vez. Asuma que los valores en las relaciones son entregados según el orden en que están almacenadas por los B+Tree. Además asuma que los valores no caben en memoria.
- b) [1 pto] Indique el costo en I/O que tendría evaluar un join con el mecanismo anterior, es decir, que revise cada tupla una única vez. Considere que R tiene R_p páginas y S tiene S_p páginas.