

Base de Datos (75.15 / 75.28 / 95.05)

Segundo Parcial Promocional

TEMA 2023141	Proc.			Fecha: 28 de junio de 2023 Padrón: _____ Apellido: _____ Nombre: _____ Cantidad de hojas: _____ <input type="checkbox"/> Aprobado <input type="checkbox"/> Insuficiente
	NoSQL			
	CRT			
	CyR			
Corrigió: Nota:				

Criterio de aprobación: El examen está compuesto por 7 ítems, cada uno de los cuales se corrige como B/B-/Reg/Reg-/M. El examen se aprueba con nota mayor o igual a 4(cuatro) y la condición de aprobación es desarrollar un ítem bien (B/B-) en al menos 3 de los 4 grupos de ejercicios (procesamiento de consultas, NoSQL, CRT, concurrencia/recuperación). Adicionalmente, no deberá haber más de dos ítems mal o no desarrollados.

1. (*Procesamiento de consultas*) Gustavo trabaja en la Dirección de Tránsito del municipio de Tandil. Cada vez que un conductor se acerca para renovar su licencia de conducir, Gustavo debe verificar que el mismo no posea ninguna multa de tránsito pendiente de pago. Para consultar el estado de deuda del conductor, Gustavo accede a una base de datos relacional con las siguientes dos tablas:

- `licencias(nro_licencia, DNI_conductor, fecha_ultima_renovacion)`
 // (740523, 41341989, 2019-03-02)
- `multas(patente_vehículo, fecha, DNI_propietario, pagada)`
 // ('AB415ZD', 2022-11-15, 41341989, False)

La base de datos también dispone de un índice secundario por `DNI_propietario` y de un índice de clustering por `pagada`, ambos sobre la tabla *Multas*.

Cuando Clemenciano Taqui llega para renovar su licencia (nro. 740523), Gustavo ejecuta la siguiente consulta en la base de datos:

```
SELECT *
FROM licencias INNER JOIN multas ON DNI_conductor=DNI_propietario
WHERE nro_licencia=740523
AND pagada IS FALSE;
```

Se pide:

- a) Proponga un plan de ejecución eficiente para esta consulta. Para ello dibuje un plan de consulta y anote sobre el mismo los métodos de acceso o algoritmos que se utilizarán en cada paso.

- b) Estime el costo del plan de ejecución que armó en el punto anterior, en términos de cantidad de accesos a bloques de disco.

Para el ejercicio considere que los índices son de tipo árbol y tienen altura 4. Además, considere para sus cálculos la siguiente información de catálogo:

LICENCIAS	MULTAS
$n(\text{Licencias}) = 30.000$	$n(\text{Multas}) = 60.000$
$B(\text{Licencias}) = 5.000$	$B(\text{Multas}) = 10.000$
$V(\text{DNI_conductor}, \text{Licencias}) = 30.000$	$V(\text{DNI_propietario}, \text{Multas}) = 15.000$

2. El Instituto Nacional de Coordinación Agrícola se ocupa de brindar asistencia a los productores rurales, contando con un extenso equipo de asesores asignados a las distintas provincias del país. Su base de datos cuenta con los siguientes esquemas de relación con datos sobre los asesores y los productores rurales:

- `Asesores(legajo, nombre, email, prov_asignada)`
- `Productores(cuit, nombre, email, hectáreas, prov_origen)`

Se quiere construir una tabla para poner en contacto a los asesores con los productores de las provincias que les fueron asignadas, a través de la siguiente operación de junta:

$$\text{Asesores} \bowtie_{\text{prov_asignada}=\text{prov_origen}} \text{Productores}$$

Estime la cardinalidad del resultado de esta junta en términos de cantidad de tuplas y en términos de cantidad de bloques, utilizando el siguiente histograma de frecuencias extraído de la información de catálogo que muestra la frecuencia de las 5 provincias principales de cada tabla. Considere además que $V(\text{prov_asignada}, \text{Asesores}) = V(\text{prov_origen}, \text{Productores}) = 23$ y que el factor de bloque de ambas tablas es de 10.

	Santa Fe	Bs.As.	Córdoba	Río Negro	Entre Ríos	otras
Asesores.prov_asignada	90	110	70	60	45	280
Productores.prov_origen	1700	2200	1100	640	720	3100

3. (NoSQL)

- a) (MongoDB) Accederemos a una base de datos en MongoDB con tweets de mayo de 2023 de todo el mundo, a fin de estimar cuáles fueron los *trending topics* del mes en Argentina. Cada tweet de esta base de datos tiene una estructura similar a la siguiente:

```

1 {
2   "created_at": "Mon May 29 20:19:24 +0000 2023",
3   "id": 1050118621198921728,
4   "id_str": "1050118621198921728",
5   "text": "Tras las #lluvias intensas de los últimos días en #Argentina,
           se observan cambios en la humedad del suelo y la vegetación del sur
           del Litoral y norte de Bs As. En colores oscuros también se
           evidencia el agua acumulada en lagunas y ríos.",
6   "user":
7     {

```

```

8         "id": 6253282,
9         "id_str": "6253282",
10        "name": "SMN Argentina",
11        "screen_name": "SMN_Argentina",
12        "followers_count": 173280,
13        "friends_count": 694
14    }
15    "entities":
16    {
17        "hashtags": ['#lluvias', '#Argentina'],
18        "user_mentions": [],
19        "media": []
20    },
21    "place":
22    {
23        "country": "Argentina",
24        "country_code": "AR"
25    },
26    "source": "Twitter Web Client"
27 }

```

Extraeremos los *trending topics* en base a los hashtags indicados en la publicación. Para ello se pide:

- 1) Escriba una consulta en MongoDB que devuelva los 5 hashtags más frecuentemente utilizados en tweets escritos desde Argentina.
 - 2) Sugiera una forma conveniente de shardear la colección a efectos de paralelizar esta consulta, e indique también si le convendría tener algún índice secundario en cada nodo de procesamiento para evitar tener que hacer un File Scan sobre todos los documentos almacenados en cada nodo. Justifique sus respuestas.
- b) (*Neo4j*) Una base de datos en Neo4j posee información sobre cada uno de los vinos que se producen en Argentina, indicando las bodegas que los producen y las cepas de uva con las que se produce cada vino, como se muestra en el siguiente ejemplo:

```

1    (v:Vino {nombre: `33 orientales`, puntaje: 37.4, precio: 315.2})
2    (b:Bodega {nombre: `El charrúa`})
3    (c1:Cepa {nombre: `Cabernet Sauvignon`})
4    (c2:Cepa {nombre: `Malbec`})
5    ...
6    (b)-[:FABRICA]-(v)
7    (v)-[:ELABORADO_CON]-(c1)
8    (v)-[:ELABORADO_CON]-(c2)

```

Los puntajes de los vinos oscilan entre 0 y 100. En particular, el vino del ejemplo anterior es fabricado por la bodega *El charrúa*, se elabora con cepas de cabernet sauvignon y malbec, y posee un puntaje de 37.4. Escriba una consulta en Neo4j que encuentre el nombre de la bodega que fabrique el vino de cepa Syrah monovarietal (es decir, que se elabora con una única cepa) de mayor puntaje en Argentina, mostrando el nombre de la bodega, el nombre del vino y su precio.

Nota: Le sugerimos utilizar la estructura `MATCH ... WHERE [NOT] EXISTS { MATCH pattern } ...` como parte de la solución.

4. (*CRT*) Considere las mismas relaciones del ejercicio 1, y escriba una expresión en *Cálculo Relacional de Tuplas* que encuentre los números de licencia de los conductores que tienen al menos dos multas impagas.

5. (Concurrencia y Recuperación)

a) (Concurrencia) Se dice que una transacción T_1 realiza una Lectura no Repetible ó *Unrepeatable Read* cuando la misma lee un ítem X , luego otra transacción T_2 escribe ese mismo ítem, y posteriormente T_1 vuelve a leer el ítem, encontrando un valor distinto al anteriormente leído. Indique si las siguientes afirmaciones sobre la Lectura no Repetible son verdaderas ó falsas. Justifique cada una de sus respuestas.

- 1) En un solapamiento recuperable puede ocurrir una Lectura no Repetible.
- 2) En un solapamiento que evita *rollbacks* en cascada puede ocurrir una Lectura no Repetible.
- 3) Aplicando el Protocolo de Lock de Dos Fases (2PL) puede ocurrir una Lectura no Repetible.
- 4) Bajo el nivel de aislamiento *Read Committed* definido en el estándar SQL puede ocurrir una Lectura no Repetible.

b) (Recuperación) Un SGBD implementa el algoritmo de recuperación UNDO/REDO con checkpoint activo. Luego de una falla, el sistema encuentra el siguiente archivo de log:

```
01 (BEGIN, T1);
02 (WRITE, T1, A, 5, 3);
03 (BEGIN, T2);
04 (WRITE, T2, B, 4, 8);
05 (WRITE, T1, C, 3, 2);
06 (BEGIN, T3);
07 (WRITE, T3, D, 15, 12);
08 (COMMIT, T3);
09 (BEGIN CKPT, {T1, T2});
10 (BEGIN, T4);
11 (WRITE, T2, D, 12, 10);
12 (WRITE, T4, E, 8, 18);
13 (COMMIT, T2);
14 (WRITE, T1, B, 8, 3);
15 (END CKPT);
16 (COMMIT, T1);
17 (WRITE, T4, C, 2, 1);
```

Explique cómo se llevará a cabo el procedimiento de recuperación, indicando qué cambios deben ser realizados en disco y en el archivo de log.