Unidad 2: Procesamiento y Optimización de Consultas.

Ejercicio 1 (Ordenación y estimación de joins)

Considerar las relaciones siguientes:

$r_1(\underline{A},B,C)$
$r_2(\underline{C},D,E)$
$r_3(\underline{E},F)$

donde r_1 tiene 1000 tuplas, r_2 tiene 1500 tuplas y r_3 750. Se pide:

a) Estimar el tamaño de la reunión $r_1 \bowtie r_2 \bowtie r_3$

b) Diseñar una estrategia eficiente para el cálculo de la reunión.

c) Suponer que no hay claves primarias, y sean

$V(C,r_1)=900$
$V(C,r_2)=1100$
$V(E,r_2)=50$
$V(E,r_3)=100$

Estimar el tamaño de $r_1 \bowtie r_2 \bowtie r_3$ y diseñar una estrategia eficiente para calcular la reunión.

Ejercicio 2

Suponer el siguiente esquema relacional de una facultad:

Asignaturas(cod_as,nom_as,creditos,cod_mat)
Mat_car(cod_mat,cod_car,creditos_min)
Carreras(cod_car,nom_car,año)

Además, se tienen los siguientes datos:

	Asignaturas	**Mat_car**	**Carreras**
Tuplas	1300	600	54
Indices 1ºs	Cod_as 2 niveles		Cod_car 1 nivel
Indices 2ºs	Cod_mat 3 niveles		
Fr	50	60	30
Observaciones	400 Materias distintas Créditos: entre 6 y 30 Distribución uniforme	400 materias distintas 54 carreras distintas Distribución uniforme	A cada nombre de carrera Le corresponden 3 carreras diferentes

Se pide:

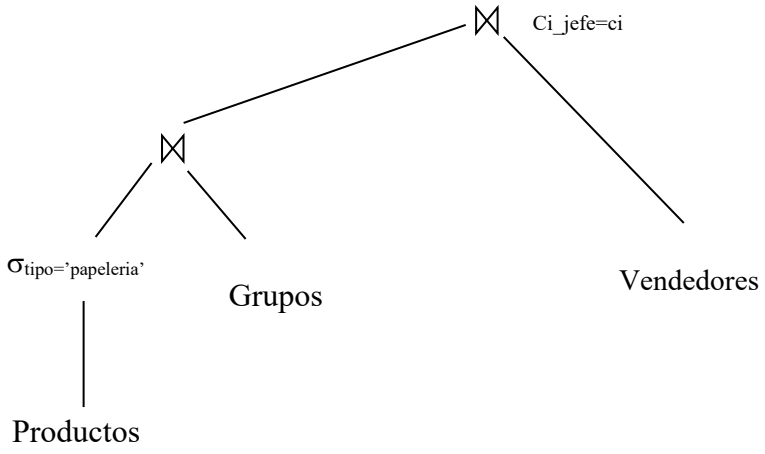
a) Obtener una consulta en el álgebra relacional para mostrar el cod_as, creditos donde se cumpla que los créditos > 10 y nom_car=”Ing. De Comp.”
b) Dar un plan lógico de la consulta, aplicando la heurística y calculando los tamaños intermedios.
c) Elegir una implementación para el primer join y calcular su coste estimado, sabiendo que cada tupla de carreras ⋈ mat_car tiene un fr=20. (factor de bloque).

Ejercicio 3

Dado el siguiente esquema relacional

Productos(tipo,<u>nroprod</u>,nrogrupo)
Grupos(<u>nrogrupo</u>,ci_jefe)
Vendedores(<u>ci</u>,nrogrupo,salario)

Y el siguiente plan lógico de consulta:



Y considerando que el resultado de la selección se debe de guardar en una tabla temporal temp1 y el resultado del primer join se debe de guardar en una tabla temporal temp2 (materializar).

a) Dar el plan físico que le parezca mejor para ese plan lógico.
b) Calcular los tamaños de temp1 y temp2.
c) Calcular el coste total del plan.

Datos:

	Productos	**Grupos**	**Vendedores**	**Temp2**
Tuplas	2000	50	10000	
Indice 1º			Ci Nivel 2	
Indice 2º	Tipo Nivel 2		Salario Nivel 3	
Fr	20	10	5	5
Observaciones	200 tipos distintos			

Ejercicio 4 (Introducción a la optimización de consultas)

Se dispone de un tamaño de bloque en disco de 2Kb, una memoria de 8 Kb y punteros de 32 bits. Se dispone de la siguiente información en el catálogo del sistema:

Empleado

NOMBRE	INIC	APELLIDO	NSS	FECHA_NAC	DIRECC	SEXO	SALARIO	NSS_SUPER	ND
50	1	100	4	8	50	1	4	4	1
			560 valores Diferen.					56 valores diferentes	5 val. dif.

Departamento

NOMBRED	NUMEROD	NSSJEFE	FECHA_INI_JEFE
50	4	4	8
	6 valores diferentes	6 valores diferentes	

Localizacion_dpto

NUMEROD	LOCALIZACIOND
4	150
6 valores diferentes	6 valores diferentes

Proyecto

NOMBREP	NUMEROP	LOCALIZACIONP	NUM_D
100	4	150	4
3590 valores diferentes	3590 valores diferentes		6 valores diferentes

Existe orden en el campo NOMBREP

Dependiente

NSSE	NOMBREDEP	SEXO	FECHA_NAC	PARENTESCO
4	150	1	8	2
560 valores diferentes	2240 valores diferentes			6 valores diferentes

Trabaja_en

NSSE	NP	HORAS
4	4	4

Hay 18500 registros.

La segunda fila de cada tabla indica la longitud de cada campo en bytes. Se pide optimizar y calcular el coste de las consultas que obtienen usando optimización heurística:

a) Nombre de las personas que dependan de trabajadores que trabajan en el depto 6. En SQL:

```
Select dependiente.nombredep
    From dependiente
inner join empleado
on dependiente.nsse=empleado.nss
where empleado.nd=6;
```

b) Nombre de las personas que dependan de los trabajadores del depto 4 y trabajan en el proyecto 7. En SQl:

```
Select dependiente.nombredep
From dependiente
Inner join empleado
        Inner join trabaja_en
        On empleado.nss=trabaja_en.nsse
On dependiente.nsse=empleado.nss
Where empleado.nd=4 and trabaja_en.np=7;
```

c) Nombre de los trabajadores y proyectos en los que trabajan los departamentos localizados en Guadalajara. En SQL:

```
Select empleado.nombre,proyecto,nombrep
From empleado
        Inner join trabaja_en
        On empleado.nss=trabaja_en.nsse
Inner join
        Proyecto
        Inner join localizacion_dpto
        On proyecto.num_d=localizacion_dpto.numerod
On trabaja_en.np=proyecto.numerop
Where localizacion_dpto.localizaciond='Guadalajara';
```

Ejercicio 5

Sea el siguiente esquema relacional representando sucursales de un banco, cuentas y clientes. Se sabe que el banco tiene sucursales en las 19 capitales del país:

Sucursales(ciudad,nombre_sucursal, fecha_inauguración, dirección, teléfono)
Cuentas(numero_cuenta, ci_titular, ci_asociado, nombre_sucursal, tipo_cta, saldo)
Clientes(ci,nombre,domicilio,teléfono)

Se tiene la siguiente información sobre las relaciones:

Sucursales:
- Ciudad: 40 bytes
- Nombre_sucursal: 40 bytes
- Fecha_inauguración: 8 bytes
- Dirección: 50 bytes
- Teléfono: 4 bytes
- Hay 40.000 registros.

Cuentas:
- Numero_cuenta: 12 bytes
- Ci_titular: 4 bytes
- Ci_asociado: 4 bytes
- Nombre_sucursal: 50 bytes
- Tipo_cta: 2 bytes

- Saldo: 4 bytes
- Los saldos se encuentran uniformemente distribuidos en el rango 1 – 1.000.000
- Hay 40.000 valores diferentes de sucursales y 100.000 valores diferentes de clientes.
- Hay 100.000 registros.

Clientes:
- Ci: 4 bytes
- Nombre: 50 bytes
- Domicilio: 50 bytes
- Teléfono: 4 bytes
- Hay 100.000 registros

El tamaño de bloque del disco es de 2 KB y la memoria tiene un tamaño de 12 KB. Se pide:

a) Construir una consulta donde se muestren los nombres y el domicilio de los clientes titulares que tengan cuentas con un salario > 500.000 euros y la ciudad de la sucursal sea distinta de Montevideo. Ordenar los resultados ascendentemente por el nombre del cliente. Expresarla en SQL y álgebra relacional. En SQL:

```
SELECT nombre,domicilio
FROM sucursales,cuentas,clientes
WHERE saldo > 500000 AND nombre_sucursal<>'Montevideo' AND
sucursales.nombre_sucursal=cuentas.nombre_sucursal AND ci_titular=ci
ORDER BY nombre ASC;
```

b) Construir un plan lógico mejorado para la consulta anterior utilizando la optimización heurística. Indicar los pasos intermedios realizados.
c) Calcular el coste total asociado al plan anterior.
d) Escribir la consulta SQL del plan lógico mejorado.
e) Indicar los cambios que se podrían introducir que podrían beneficiar el coste total del plan anterior.

Ejercicio 6

Considerar una parte de la base de datos de una universidad que tiene las siguientes 4 relaciones:

ESTUDIANTE (DNI, Nombre)
MATRICULA (DNI, ID_Asig)
NOTAS (DNI, ID_Asig, Nota)
ASIGNATURA (ID_Asig, Nombre , Profesor)

Se tiene la siguiente información sobre las relaciones:

- ESTUDIANTE, contiene 40.000 tuplas.
- MATRICULA, guarda las asignaturas en las cuales los alumnos se encuentran actualmente matriculados (en curso), asumiendo 10 asignaturas por estudiante en el año académico.
- NOTAS, guarda la nota de las asignaturas que el alumno ha aprobado, teniendo un número medio de 15 asignaturas aprobadas por alumno, y cuya nota puede ser A, B, C ó D, estando éstas distribuidas respectivamente de la siguiente manera: 50%, 25%, 15% y 10%. Todas las asignaturas están registradas en la tabla notas.
- ASIGNATURA, representa las asignaturas ofrecidas por la universidad, teniendo 5000 diferentes.

Los campos que son clave primaria tienen una longitud de 20 bytes y los restantes una longitud de 40 bytes. El bloque de disco tiene un tamaño de 1 K. Se da la condición de mínima memoria suficiente para cada operación.Se pide:

a) Considerar la consulta "Obtener una lista de los nombres de los estudiantes que no han obtenido un grado de "C" o mayor en la signatura "BDA". Expresarla en SQL y álgebra relacional. En SQL:

   Select nombre
   From estudiante,nota,asignatura
   Where nota <'C' and asignatura.nombre='BDA' and estudiante.dni=nota.dni and nota.id_asig=asignatura.id_asig;

b) Diseñar un plan lógico y físico para la consulta anterior utilizando la optimización heurística, indicando el proceso seguido.
c) Calcular el coste total asociado al plan anterior asumiendo el peor caso para las operaciones de reunión, sabiendo que el resultado se tiene que escribir en un fichero.
d) Indicar los cambios que se podrían introducir que podrían beneficiar el coste total del plan anterior.)

Ejercicio 7

Se define la anti semi reunión $T = R \overline{\ltimes} S$, como la lista de tuplas de R que ***no*** concuerdan con ninguna tupla de S, en los atributos comunes de R y de S. Se pide:

a) Dar una expresión de álgebra relacional equivalente a T
b) Estimar el tamaño (en tuplas) de T

Ejercicio 8

Considere las siguientes relaciones de una base de datos relacional, donde las claves primarias y ajenas tienen una longitud de 4 bytes y las demás 20 bytes. Por el momento no hay índices disponibles ni ordenación de las tablas. También se sabe que la distribución por apellidos es la siguiente:

- A-E , 25 %
- F-J , 20 %
- K-O , 20 %
- P-T , 20 %
- U-Z , 15 %

PROF(ci,nombre,apellido,datos_personales)

Contiene los datos personales de los profesionales que trabajan. Hay 1000 registros.

DEPTO(cod_d, nombre_d, ci_jefe)

Contiene los departamentos en los cuales trabajan los profesionales y además el jefe del departamento que será un profesional. Hay 20 departamentos cada uno con un jefe diferente.

PROYECTOS(n_proyi, nombre_pi,ci_director,descripción_pi)

Contiene los proyectos de investigación en los que se trabajan, así como el director del proyecto, que será un profesional. Hay 2000 proyectos de investigación y 500 directores de proyecto.

GRUPO_INV(n_proyi,ci_inv)

Contiene los profesionales que trabajan en los proyectos de investigación. Hay 10.000 registros y todos trabajan en los proyectos.

Sabiendo que se dispone de una memoria de 24 Kb para cada operación y el bloque de disco es de 2 Kb, se pide:

a) Escribir en SQL una consulta que obtenga los nombres de los proyectos en los que trabajan las personas que se apellidan Gutiérrez y que además no sean directores de proyecto. Recordatorio (UNION es la unión de dos relaciones, EXCEPT es la diferencia de dos relaciones e INTERSECT es la intersección de dos relaciones). En SQL:

```
SELECT nombre_pi
FROM proyectos, grupo_inv, prof
WHERE apellido = 'Gutierrez' AND proyectos.n_proyi = grupo_inv.n_proyi AND ci_inv = ci
        EXCEPT
SELECT nombre_pi
FROM proyectos, prof
WHERE apellido = 'Gutierrez' AND ci_director = ci;
```

b) Optimizar la consulta anterior, indicando el plan lógico y físico final que se aplicaría.
c) Calcular el coste asociado al plan anterior.
d) Que cambios serían convenientes introducir para reducir el coste asociado al plan anterior.

Ejercicio 9

Un operador unario se dice que es **idempotente** si *para todas las relaciones R, se cumple que* ***f (f (R)) = f (R)***. Esto es, aplicar ***f*** más de una vez es lo mismo que aplicarla sólo una vez. *Justifique* cual de los siguientes ejemplos es idempotente: bien demuéstrelo o bien proporcione un contraejemplo:

(a) ρ_S (b) σ_C

(c) π_L (d) τ_A (ordenar por A)

Ejercicio 10 (Estimación de tuplas y costes)

Considere un optimizador de consultas que usa histogramas. En particular, se conoce la siguiente información sobre un atributo A de la relación R. El atributo A es de tipo entero.

- Existen 100 tuplas con un valor de A comprendido entre 1 y 10. En este rango hay 8 valores distintos de A.
- Existen 200 tuplas con un valor de A comprendido entre 11 y 20. En este rango hay 5 valores distintos de A.
- Existen 300 tuplas con un valor de A comprendido entre 21 y 30. En este rango hay 10 valores distintos de A.

- Existen 400 tuplas con un valor de A comprendido entre 31 y 40. En este rango hay 10 valores distintos de A.

Se pide:

(a) Considere la consulta $\sigma_{A=7}$ (R). ¿Cuántas tuplas son de esperar en la respuesta, suponiendo que los valores están distribuidos uniformemente sobre los posibles valores V (R,A)?

(b) Considere la consulta $\sigma_{A=17}$ (R). ¿Cuántas tuplas son de esperar en la respuesta, suponiendo que los valores están distribuidos uniformemente sobre los posibles valores del dominio?

(c) Considere la consulta $\sigma_{A>17}$ (R). ¿Cuántas tuplas son de esperar en la respuesta, suponiendo que los valores están distribuidos uniformemente sobre los posibles valores del dominio?

(d) Considere la consulta R ⋈ S, donde R tiene los atributos R (A,B,C) y S tiene los atributos S (A,D,E). Suponga que S tiene el mismo número de tuplas que R, y que el atributo A de S tiene el mismo histograma que A tiene en R. Suponiendo que los valores están uniformemente distribuidos sobre los posibles valores V(A,R), ¿Cuántas tuplas son de esperar en la respuesta?

Ejercicio 11 (Estimación de tuplas y costes)

Considere tres relaciones R_1 (A,B,C), R_2 (C,D,E), R_3 (E,F,A). Se desea estimar el coste (numero de I/Os de disco) al responder a la consulta $R_1 \bowtie R_2 \bowtie R_3$. Se tiene la siguiente información:

- R_1 contiene 10.000 tuplas, almacenadas contiguamente en 1.000 bloques
- R_2 contiene 20.000 tuplas, almacenadas contiguamente en 2.000 bloques
- R_3 contiene 30.000 tuplas, almacenadas contiguamente en 3.000 bloques
- Si se calculara el resultado parcial $R_1 \bowtie R_2$, contendría 1.500 tuplas, que almacenadas en disco ocuparían 300 bloques.
- Si se calculara el resultado parcial $R_2 \bowtie R_3$, contendría 1.000 tuplas, que almacenadas en disco ocuparían 200 bloques.
- Si se calculara el resultado parcial $R_1 \bowtie R_3$, contendría 500 tuplas, que almacenadas en disco ocuparían 100 bloques.

Para esta cuestión considere solamente un algoritmo de reunión por asociación (*hash-join*).
Suponga que hay suficiente memoria para ejecutar la reunión.

(a) *Justifique brevemente* cual es el mejor plan para calcular la consulta.

(b) ¿Cuántas operaciones de I/O se necesitan para este mejor plan?. Explique brevemente cómo ha calculado su respuesta.

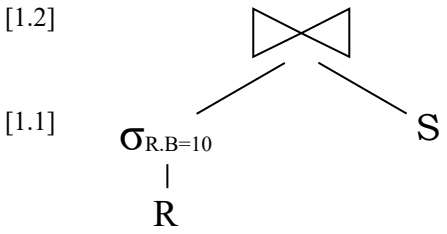
Ejercicio 12 (Estimación de tuplas y costes)

El procesador de consultas va a decidir como ejecutar la siguiente consulta en las relaciones R (A,B) y S (B,C):

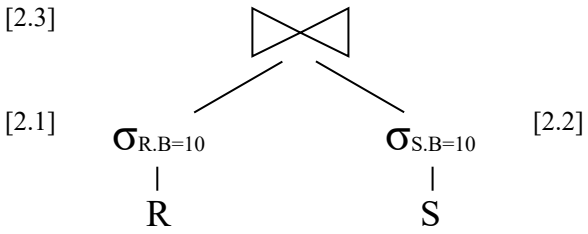
```
SELECT * FROM R, S WHERE R.B = S.B AND S.B = 10
```

Para ello, el procesador va a considerar los dos siguientes planes:

Plan 1



Plan 2



La relación R contiene 60.000 tuplas, con 10 tuplas en cada bloque de disco, y V (R,B) = 12. Similarmente, S contiene 30.000 tuplas, con 30 tuplas en cada bloque de disco, y V (S,B) = 5.

(a) Considere un modelo que estima el coste de un plan de consulta mediante el número total de tuplas *intermedias y finales* producidas por los operadores. ¿Cuáles son los tamaños (número de tuplas) de los resultados intermedios y resultados finales de cada plan? Suponga que los valores en cada relación están distribuidos entre todos los posibles valores de V (B,R) ¿Cuáles son los costes totales de los planes 1 y 2? *Justifique* cuál es el mejor plan.

(b) Suponga ahora lo siguiente:
- La reunión del Plan 1 se implementa por asociación (*hash-join*)
- El resultado intermedio de la operación [1.1] no se escribe en disco
- Los cajones se almacenan en disco
- El resultado final se guarda sólo en memoria
- Hay suficiente memoria para el algoritmo de reunión por asociación

¿Cuántos accesos a disco requiere el Plan 1?

(c) Con las mismas suposiciones (idénticas, más la de que el resultado de [2.1] y [2.2] no se almacena en disco), ¿Cuántos accesos a disco requiere el Plan 2?

(d) *Justifique* los resultados

Ejercicio 13

Considere el siguiente esquema donde los campos clave están subrayados.

```
Suministradores (sNIF, sNombre, sDireccion), hay 100
registros.
Piezas (pID, pNombre, pColor), hay 2000 registros.
Catalogo (sNIF, pID, cCoste), hay 4000 registros.
```

Suponga que cada campo clave tiene una longitud de 8 bytes y el que no es clave 20 bytes. El tamaño del bloque es de 1024 bytes.

Se desea construir la consulta "Hallar el NIF de los Suministradores que proporcionan **todas** las Piezas"

(a) Escriba la consulta en el álgebra relacional. En SQL: SELECT nif FROM (SELECT * FROM (SELECT nif FROM Suministradores) as A, (SELECT id FROM Piezas) as B) EXCEPT SELECT nif,id FROM C) as C EXCEPT SELECT nif FROM Suminitradores;

(b) Haga un diagrama con el árbol resultante. Optimícelo.

(c) Calcular el coste asociado al plan final, **asumiendo el peor caso**.

(d) Indicar como se podría mejorar el plan anterior.

Ejercicio 14

Considere tres relaciones R_1 (A,B,C), R_2 (C,D,E), R_3 (E,F,A). Se desea estimar el coste (numero de I/Os de disco) al responder a la consulta $R_1 \bowtie R_2 \bowtie R_3$. Se tiene la siguiente información:

- R_1 contiene 32.000 tuplas, almacenadas contiguamente en 3.200 bloques
- R_2 contiene 8.000 tuplas, almacenadas contiguamente en 800 bloques
- R_3 contiene 16.000 tuplas, almacenadas contiguamente en 1.600 bloques
- Si se calculara el resultado parcial $R_1 \bowtie R_2$, contendría 3.200 tuplas, que almacenadas en disco ocuparían 400 bloques.
- Si se calculara el resultado parcial $R_2 \bowtie R_3$, contendría 1.600 tuplas, que almacenadas en disco ocuparían 200 bloques.
- Si se calculara el resultado parcial $R_1 \bowtie R_3$, contendría 800 tuplas, que almacenadas en disco ocuparían 100 bloques.

Para esta cuestión considere solamente un algoritmo de reunión por asociación (*hash-join*).
Suponga que hay suficiente memoria solamente para efectuar las operaciones de reunión, y que los posibles resultados intermedios se materializan.

(a) *Justifique* cual es el mejor plan para calcular la consulta.

(b) ¿Cuántas operaciones de I/O se necesitan para este mejor plan?. *Explique* brevemente cómo ha calculado su respuesta.

(c) Calcule cual sería el valor mínimo de bloques de memoria que permite, según el enunciado, que "*... hay suficiente memoria solamente para efectuar las operaciones de reunión...*". *Justifíquelo*.

(d) Suponga ahora que puede disponer de la memoria adicional que desee. Calcule cual sería el nuevo coste de la operación de reunión, y el número de bloques mínimos necesarios en esta situación. *Justifíquelo*.

Ejercicio 15 (Estimación de tuplas y costes)

Considerar la siguiente relación R, donde el tamaño de bloque es de 2 K, y cada campo ocupa 20 bytes. Suponer inicialmente que ***no*** se tienen estadísticas sobre la tabla.

MARCA	MODELO	PUERTAS
HONDA	CIVIC	2
HONDA	ACCORD	2
HONDA	SUV	2
FORD	TAURUS	4
FORD	MUSTANG	4
FORD	SUV	4
GM	CHEVY	2
GM	BUICK	2
GM	SUV	2

1. ¿Se podría estimar el número de tuplas esperado en la consulta $\sigma_{MARCA=HONDA}$ (R)? Si es así, determinar ese valor, si no es así, justificarlo

2. ¿Qué estadísticas podría tener almacenada la base de datos en el catálogo del sistema si se decide recolectarlas? ¿Cuáles serían los valores de estas estadísticas?

3. Considerar de nuevo la consulta $\sigma_{MARCA=HONDA}(R)$. ¿Cuál sería el número de tuplas estimado por el optimizador de consultas utilizando las estadísticas recolectadas?

4. Considerar la consulta $\sigma_{MARCA=HONDA \wedge MODELO=SUV}(R)$. ¿Cuál sería el número de tuplas estimado por el optimizador de consultas utilizando las estadísticas?

5. Considerar la consulta $\sigma_{MARCA=HONDA \wedge PUERTAS=2}(R)$. ¿Cuál sería el número de tuplas estimado por el optimizador de consultas utilizando las estadísticas?

6. De los resultados de las consultas (4) y (5) hay una que se acerca más a la realidad. ¿Cuál es, y a qué puede ser debido?

Ejercicio 16 (Tipo Examen)

Considere el siguiente esquema de una base de datos que almacena información sobre usuarios que han comprado canciones en YouClip:

```
USUARIO (DNI, nombre, dirección, teléfono)
COMPRA (DNI, IDCANCION, fecha, coste)
CANCION (IDCANCION, titulo, formato, duración)
```

Se tiene la siguiente información sobre las tablas referentes al año 2006:

- USUARIO, contiene 50.000 registros. En media cada usuario ha comprado una canción
- COMPRA guarda el usuario que ha comprado una canción en tal fecha y con un coste total asociado. La distribución de las compras a lo largo del año viene dada por el siguiente porcentaje: 1er trimestre 30 %, 2º trimestre 50 %, 3er trimestre 10% y 4º trimestre 10%. Sólo se han comprado 20.000 canciones diferentes.
- CANCION, contiene 100.000 registros, donde hay 20 formatos diferentes de canciones

Los campos que son clave tienen una longitud de 20 bytes y los restantes una longitud de 40 bytes. El bloque de disco tiene un tamaño de 1 K. En cuanto a la memoria, se puede utilizar como máximo 12 K para cada operación individual y es posible encauzar una de las ramas en las operaciones con 2 operandos.

1. Dar una expresión del algebra relacional para la consulta “Obtener el nombre de los clientes que han comprado alguna canción en formato WAV o MP3 en el cuarto trimestre del año”. En SQL: SELECT nombre FROM usuario NATURAL JOIN compra NATURAL JOIN canción WHERE fecha>=’01-10-2006’ AND (formato=’MP3’ OR formato=’WAV’);

2. Diseñar un plan lógico y un plan físico para la consulta anterior utilizando la optimización heurística, indicando el proceso seguido

3. Calcular el coste total asociado al plan anterior bajo las condiciones comentadas anteriormente

4. Indicar los cambios que se podrían introducir que podrían beneficiar el coste total del plan anterior, haciendo una **valoración aproximada** del beneficio

Ejercicio 17

Aplique la regla heurística de "descender al máximo posible" ciertos operadores y rescriba, para optimizar, las siguientes expresiones de las relaciones R(a,b), S(b,c) y T(a,d):

5. $\pi_{a,c}(\sigma_{b<5}(R \bowtie S))$
6. $\pi_{b,c}(\sigma_{(a<b)\wedge(a<3)}(R \bowtie S))$
7. $\pi_{d}(\sigma_{c<5}(R \bowtie S \bowtie T))$

Ejercicio 18 (Tipo Examen)

Considere las siguientes relaciones de una base de datos relacional, donde todas las claves (candidatas, ajenas, etc.) tienen una longitud de 4 bytes y las demás 20 bytes. Por el momento no hay índices disponibles ni ordenación de las tablas. Se disponen de las siguientes tablas con las siguientes estadísticas:

PROF (ci_inv, nombre, apellido, datos_personales). Contiene los datos personales de los profesionales que trabajan en investigación. Hay 1.000 registros y se sabe que hay 20 apellidos diferentes.

DEPTO (cod_depto, nombre_depto, ci_jefe). Contiene los departamentos en los cuales trabajan los profesionales y además el jefe del departamento (ci_jefe) que será un profesional. Hay 20 departamentos diferentes y por lo tanto 20 jefes diferentes.

PROYECTOS (ni_proy, nombre_pi, ci_director, descripción_pi). Contiene los proyectos de investigación en los que se trabajan así como el director del proyecto (ci_director), que será un profesional. Hay 30 proyectos de investigación y por lo tanto 30 directores diferentes.

GRUPO_INV (ni_proy, ci_inv). Contiene los profesionales que trabajan en los proyectos de investigación. Hay 5.000 registros y todos los proyectos y investigadores se encuentran registrados.

Sabiendo que el bloque de disco es de 2KB, hay memoria suficiente para encauzar lo que se pueda, que la salida hay que grabarla en un fichero y que sólo hay 3 bloques de memoria para realizar **cualquier operación binaria**, se pide:

1. Dar una expresión del álgebra relacional para la consulta: “Obtener el nombre de los profesionales que se apelliden Fernández y que **no** sean directores de proyecto". En SQL: SELECT nombre FROM prof WHERE apellido=’Fernández’ EXCEPT SELECT nombre FROM prof NATURAL JOIN proyectos WHERE apellido=’Fernández’;
2. Optimizar la consulta anterior, indicando ***claramente***: (a) el plan lógico de la expresión anterior, (b) la heurística considerada, y (c) el plan físico final que se aplicaría.
3. Calcular el coste asociado al plan anterior.
4. Justificar que cambios serían convenientes introducir para reducir el coste asociado al plan anterior, y **calcular** dicho ahorro.

Ejercicio 19 (Tipo Examen)

Considerar una parte de la base de datos de un concesionario que guarda la relación entre los clientes del concesionario, los automóviles que vende y las compras de cada cliente.

CLIENTE (DNI, Nombre,Dirección,Telefono) , donde DNI contiene el DNI del cliente, Nombre el nombre del cliente, Dirección la dirección de residencia y Teléfono el teléfono.
COMPRA(DNI, BASTIDOR, Fecha,Coste), DNI contiene el DNI del cliente que compró el automóvil, Bastidor el número de bastidor del vehículo , Fecha la fecha de compra y Coste el precio total pagado por el cliente.
AUTOMOVIL (BASTIDOR, Proveedor, Modelo,Precio), donde Bastidor contiene el número de bastidor del vehículo, Proveedor el nombre de la empresa que lo fabrica, Modelo el modelo del vehículo y precio el precio de venta del vehículo.

Se tiene la siguiente información sobre las relaciones referentes al año 2005:

- CLIENTE, contiene 50.000 registros y en media cada cliente ha comprado un automóvil.
- COMPRA guarda el cliente que ha comprado un coche en tal fecha y con el coste total asociado. La distribución de las compras a lo largo del año viene dada por el siguiente porcentaje: 1er cuatrimestre 30 %, 2º cuatrimestre 50 % y 3er cuatrimestre 20%.
- AUTOMOVIL, en la base de datos hay 100.000 registros, donde hay 500 proveedores del automóvil.

Los campos que son clave primaria tienen una longitud de 20 bytes y los restantes una longitud de 40 bytes. El bloque de disco tiene un tamaño de 1 K. Además se sabe que existe un índice de árbol B+ sobre el atributo DNI de la tabla CLIENTE con una altura de 3, y además sobre el atributo PROVEEDOR de la tabla AUTOMOVIL existe un índice de árbol B+ de nivel 4. Además se sabe que hay memoria para encauzar lo que sea posible y que cada una de las operaciones de join tiene una memoria máxima de 5 bloques asignada. Se pide:

1. Considerar la consulta “Obtener el nombre de los clientes que han comprado un automóvil del proveedor Renault o SEAT en el tercer cuatrimestre del año”. En SQL: SELECT nombre from automóvil NATURAL JOIN compra NATURAL JOIN cliente WHERE fecha=’3er cuatrimestre AND (proveedor=’Renault’ OR proveedor=’Seat’); Expresarla en álgebra relacional.

2. (a) Describir inicialmente un árbol de consulta equivalente. (b) Proponer los criterios particulares de optimización para este árbol, siendo especialmente claros en los algoritmos y consideraciones aplicadas. (c) Describir finalmente el árbol optimizado resultante.

3. Calcular los costes asociados al árbol optimizado anterior.

4. Indicar las mejoras que se podrían realizar para ahorrar coste. En caso de haberlas, calcular ese ahorro. Si no hay mejoras, especificarlo.

Ejercicio 20(Tipo Examen)

Considerar una parte de la base de datos de un concesionario que guarda la relación entre los clientes del concesionario, los automóviles que vende y las compras de cada cliente.

CLIENTE (ID_CLIENTE, DNI, Nombre, Dirección, Telefono), donde ID_CLIENTE contiene el identificador del cliente siendo un número consecutivo empezando por 0, DNI el DNI del cliente, Nombre el nombre del cliente, Dirección la dirección de residencia y Teléfono el teléfono.
COMPRA(ID_CLIENTE, ID_AUTO, Fecha, Coste), DNI contiene el DNI del cliente que compró el automóvil, Bastidor el número de bastidor del vehículo , Fecha la fecha de compra y Coste el precio total pagado por el cliente.
AUTOMOVIL (ID_AUTO, Bastidor, Proveedor, Modelo, Precio), donde ID_AUTO contiene el identificador del vehículo siendo un número consecutivo empezando por 0, Bastidor contiene el número de bastidor del vehículo, Proveedor el nombre de la empresa que lo fabrica, Modelo el modelo del vehículo y precio el precio de venta del vehículo.
Se tiene la siguiente información sobre las relaciones referentes al año 2005:

- CLIENTE, contiene 60.000 registros.
- COMPRA, contiene 90.000 registros y guarda el cliente que ha comprado un coche en tal fecha y con el coste total asociado. La distribución de las compras a lo largo del año viene dada por el siguiente porcentaje: 1er cuatrimestre 40 %, 2º cuatrimestre 30 % y 3er cuatrimestre 30%.Cada cliente almacenado en compra ha comprado un coche
- AUTOMOVIL, en la base de datos hay 120.000 registros, donde hay 500 proveedores del automóvil. Se sabe que la tabla se encuentra ordenada por el campo Proveedor.

Los campos que son clave primaria tienen una longitud de 20 bytes y los restantes una longitud de 40 bytes. El bloque de disco tiene un tamaño de 1 K. Además se sabe que hay memoria para encauzar lo que sea posible y que hay una memoria mínima suficiente para realizar por cualquier algoritmo las operaciones binarias. La función de asociación que se puede aplicar asociada a los campos numéricos es X mod 100, siendo X el valor del campo numérico Se pide:

a) Considerar la consulta “Obtener el identificador de todos los clientes que han comprado un automóvil del proveedor SEAT y que además no hayan comprado ningún otro automóvil en el tercer cuatrimestre”. No utilizar el operador ≠. En SQL: SELECT id_cliente FROM automóvil NATURAL JOIN compra WHERE proveedor=’Seat’ EXCEPT Select id_cliente FROM compra WHERE fecha=’3er cuatrimestre’;

b) (1) Describir inicialmente un árbol de consulta equivalente. (2) Proponer los criterios particulares de optimización para este árbol, siendo especialmente claros en los algoritmos y consideraciones aplicadas. (3) Describir finalmente el árbol optimizado resultante.

c) Calcular los costes asociados al árbol optimizado anterior.

d) Determinar el número mínimo de bloques de memoria necesarios para poder reducir al máximo el coste estimado del árbol.