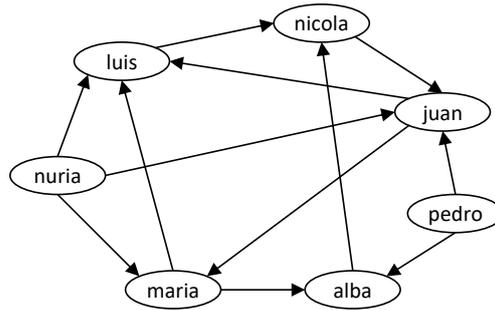


# Estructuras de Datos

## Grado en Ingeniería Informática, 2º curso

### Ejercicios

- Escribir en SQL las sentencias necesarias para crear una base de datos que almacene información de una red social asimétrica (tipo "follow") en la que los usuarios tienen un nick, nombre y email.
  - Definir adecuadamente las claves primarias y restricciones de cada tabla.
  - Insertar los datos necesarios para almacenar la siguiente red social (sólo se indican los nicks; definir libremente el resto de datos):



- Los seguidores comunes de luis y maria.
  - Los seguidores de "grado dos" de nicola.
  - Reescribir las consultas anteriores considerando que la red fuese simétrica (la relación de "seguir" siempre es mutua).
- Escribir en SQL las sentencias necesarias para almacenar la información sobre vuelos comerciales incluyendo los siguientes datos:
  - Aeropuertos con código (tres letras, único) y ciudad (una ciudad puede tener varios aeropuertos).
  - Vuelos con número (único), origen y destino (códigos de aeropuerto), hora de salida, nº de plazas (suponiendo que siempre fuese el mismo modelo de avión), y línea aérea que los opera.
  - Líneas aéreas con nombre y abreviatura (única).
  - Pasajeros con DNI y nombre.
  - Reservas de vuelos por pasajeros en una fecha dada, con su precio.Poblar las tablas con algunos datos.
- Escribir en SQL las sentencias de actualización que apliquen los siguientes cambios en la base de datos del ejercicio anterior:
  - Anular todas las reservas de los vuelos que salen de Madrid.
  - Reasignar a Iberia todos los vuelos de British Airways que salen de Madrid.
  - Duplicar el precio de las reservas de easyjet del mes de agosto 2010.
- Escribir en SQL las consultas que permitan obtener la siguiente información en la base de datos del ejercicio 3:
  - Vuelos que salen de París.
  - Todos los vuelos entre Madrid y París con salida antes de las 12:00.
  - Nombre de pasajeros que vuelan de Londres a París, indicando la fecha.
  - Nombre de pasajeros que vuelan entre Londres y París, en cualquier sentido.
  - Pasajeros que hacen un trayecto de ida y vuelta en el mismo día.

6. Escribir en SQL las consultas que permitan obtener la siguiente información en la base de datos del ejercicio 3:
  - a. Líneas que no tienen vuelos con salida desde Londres.
  - b. Vuelos completos, indicando la fecha.
  - c. Vuelos vacíos (sin reservas) el 1 de enero de 2018.
  - d. Líneas que sólo operan vuelos con salida o llegada a Madrid.
  
7. Escribir en SQL las consultas que permitan obtener la siguiente información en la base de datos del ejercicio 1:
  - a. Usuarios que no siguen a nadie.
  - b. Usuarios sin seguidores.
  - c. Usuarios a quienes sigue todo el mundo.
  
8. Escribir en SQL las consultas que permitan obtener la siguiente información en la base de datos del ejercicio 3:
  - a. Aeropuerto con más tráfico (contando salidas y llegadas).
  - b. Líneas aéreas ordenadas por el nº total de reservas en los vuelos que operan.
  - c. Nombre de la ciudad desde la que sale el primer vuelo del día.
  - d. Gasto total promedio por pasajero.
  - e. Facturación total por línea aérea y aeropuerto de salida.
  
9. Dado el estado de la siguiente base de datos de una empresa:

PROYECTO			
NombreProyecto	NumProyecto	UbicacionProyecto	NumDptoProyecto
ProductoX	1	Valencia	5
ProductoY	2	Sevilla	5
ProductoZ	3	Madrid	5
Computación	10	Gijón	4
Reorganización	20	Madrid	1
Comunicaciones	30	Gijón	4

SUBORDINADO				
DniEmpleado	NombSubordinado	Sexo	FechaNac	Relación
333445555	Alicia	M	05-04-1986	Hija
333445555	Teodoro	H	25-10-1983	Hijo
333445555	Luisa	M	03-05-1958	Esposa
987654321	Alfonso	H	28-02-1942	Esposo
123456789	Miguel	H	04-01-1988	Hijo
123456789	Alicia	M	30-12-1988	Hija
123456789	Elisa	M	05-05-1967	Esposa

EMPLEADO									
Nombre	Apellido1	Apellido2	Dni	FechaNac	Dirección	Sexo	Sueldo	SuperDni	Dno
José	Pérez	Pérez	123456789	01-09-1965	Eloy I, 98	H	30000	333445555	5
Alberto	Campos	Sastre	333445555	08-12-1955	Avda. Rios, 9	H	40000	888665555	5
Alicia	Jiménez	Celaya	999887777	12-05-1968	Gran Vía, 38	M	25000	987654321	4
Juana	Sainz	Oreja	987654321	20-06-1941	Cerquillas, 67	M	43000	888665555	4
Fernando	Ojeda	Ordóñez	666884444	15-09-1962	Portillo, s/n	H	38000	333445555	5
Aurora	Oliva	Avezuela	453453453	31-07-1972	Antón, 6	M	25000	333445555	5
Luis	Pajares	Morera	987987987	29-03-1969	Enebros, 90	H	25000	987654321	4
Eduardo	Ochoa	Paredes	888665555	10-11-1937	Las Peñas, 1	H	55000	NULL	1

DEPARTAMENTO			
NombreDpto	NumeroDpto	DniDirector	FechaIngresoDirector
Investigación	5	333445555	22-05-1988
Administración	4	987654321	01-01-1995
Sede Central	1	888665555	19-06-1981

LOCALIZACIONES_DPTO	
NumeroDpto	UbicacionDpto
1	Madrid
4	Gijón
5	Valencia
5	Sevilla
5	Madrid

Mostrar el resultado de las siguientes sentencias SQL, y en su caso el estado de las tablas modificadas (suponer que los comandos de SQL siempre se aplican a la BD original):

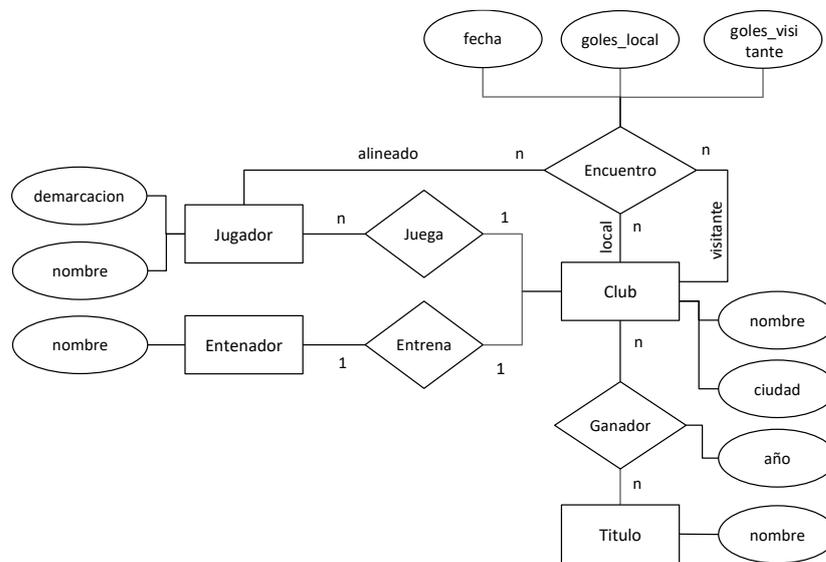
- a. DELETE FROM EMPLEADO WHERE dni='123456789';

- b. DELETE FROM EMPLEADO WHERE Apellido1='Cabrera';
  - c. ALTER TABLE EMPLEADO ADD COLUMN Trabajo VARCHAR (12);
  - d. ALTER TABLE EMPLEADO DROP COLUMN dirección CASCADE;
  - e. INSERT INTO EMPLEADO VALUES ('Ricardo', 'Roca', 'Flores', '653298653', '1962-12-30', 'Los Jarales, 47', 'H', 37000, '653298653', 4);
  - f. INSERT INTO EMPLEADO (Nombre, Apellido1, Dno, Dni) VALUES ('Ricardo', 'Roca', 4, '653298653');
  - g. UPDATE PROYECTO SET UbicaciónProyecto='Valencia', NumDptoProyecto = 5 WHERE NumProyecto=10;
  - h. SELECT FechaNac, Dirección FROM EMPLEADO WHERE Nombre='José' AND Apellido1='Pérez' AND Apellido2='Pérez';
  - i. SELECT Nombre, Apellido1, Dirección FROM EMPLEADO, DEPARTAMENTO WHERE NombreDpto='Investigación' AND NumeroDpto=Dno;
  - j. SELECT NumProyecto, NumDptoProyecto, Apellido1, Dirección, FechaNac FROM PROYECTO, DEPARTAMENTO, EMPLEADO WHERE NumDptoProyecto=NumeroDpto AND DniDirector=Dni AND UbicacionProyecto='Gijon';
  - k. SELECT Nombre, Apellido1, Dirección FROM (EMPLEADO JOIN DEPARTAMENTO ON Dno=NumeroDpto) WHERE NombreDpto='Investigación';
10. Diseñar un diagrama Entidad Relación que describa en detalle las entidades, atributos, relaciones y claves para almacenar en una base de datos relacional la siguiente información:
- a. Películas, con título, nacionalidad, género, reparto, director, guionista, fecha de estreno.
  - b. Actores, con nombre, género, y películas en las que han actuado (indicando el personaje que interpretan).
  - c. Directores, con nombre y películas dirigidas.
  - d. Guionistas, con nombre y películas cuyo guión han escrito.
- Indicar con detalle las restricciones de cardinalidad, así como las claves que pudiese haber.
- Para simplificar, considerar inicialmente:
- Que los títulos de las películas y los nombres de las personas no se repiten.
  - Que nadie puede ser a la vez actor y director o guionista, sino sólo una de las tres cosas.
11. Refinar el diseño de la pregunta anterior con las siguientes elaboraciones:
- a. Eliminando la última restricción, es decir, teniendo en cuenta que un cineasta puede trabajar como actor, director y/o guionista en diferentes momentos de su carrera (o incluso simultáneamente).
  - b. Añadiendo información sobre los óscars recibidos, reflejando quién o qué lo recibe (actor, película, director, etc.), la modalidad (mejor película, mejor guión, mejor actor protagonista, etc.), la fecha, y las circunstancias (p.e. mejor actor por un papel determinado, mejor director por una película concreta, etc.).
12. Convertir en tablas el modelo diseñado en las dos preguntas anteriores. Incluir algún registro de ejemplo en cada tabla.
13. Describir las estructuras del ejercicio anterior en el modelo relacional.
- a. Describir con detalle los esquemas y elementos del modelo en notación conjuntista.
  - b. Escribir las sentencias SQL necesarias para implementar los esquemas en una base de datos relacional.
14. Una cadena de librerías requiere una aplicación de gestión de negocio que cubra la gestión del catálogo, existencias, ventas, y recursos humanos. Diseñar un modelo Entidad Relación para el desarrollo de una base de datos que dé soporte a esta aplicación, gestionando la siguiente información, con los datos que se indican:
- a. Libros a la venta: título, autor(es), editorial, ISBN y precio.
  - b. Editoriales de los libros: nombre, CIF, teléfonos de contacto.
  - c. Autores de los libros: nombre (se supone que el nombre no se repite, o que a la librería no le importa).

- d. Empleados de la librería: nombre, NIF, sueldo, y tienda en la que está destinado.
- e. Tiendas de la cadena: dirección postal, encargado (uno de los empleados cumple esta función), y nº de ejemplares de cada libro disponibles en la tienda.
- f. Ventas: libro, fecha de venta, tienda donde se ha vendido el libro, y empleado que ha hecho la operación.

Detallar los aspectos de cardinalidad, participación total o parcial en las relaciones, claves primarias, entidades débiles si las hubiera, y no incluir más datos que los que estrictamente se indican.

- 15. Convertir a esquemas de modelo relacional las estructuras del ejercicio anterior. Y convertir a SQL.
- 16. El siguiente diagrama E/R describe estructuras de base de datos para almacenar información sobre clubes, jugadores y resultados de una temporada de la liga de fútbol (para simplificar se supone que los clubes no cambian de entrenador, ni los jugadores cambian de equipo, a mitad de temporada):



Convertir este diseño en esquemas relacionales, indicando cuáles serían las claves primarias y foráneas. Nota: la demarcación de un jugador puede ser 'portero', 'defensa', 'centrocampista', 'delantero'. Supondremos para simplificar que los nombres de entrenadores y jugadores no se repiten.

- 17. Escribir en SQL las estructuras del ejercicio anterior.
- 18. Diseñar un modelo E/R para las estructuras del ejercicio 1.
- 19. Diseñar un modelo E/R para las estructuras del ejercicio 3.
- 20. Diseñar un modelo E/R para las tablas del ejercicio 9.
- 21. Dados los siguientes esquemas:

EQUIPO (Nombre, Estadio, Presidente, Entrenador, Num\_socios)  
 JUGADOR (Nombre, Dorsal, Posición, Nacionalidad, Nombre\_equipo, Entrenador)  
 VUELO (Nvuelo, Hora, Origen, Destino, Ciudad\_origen, Ciudad\_destino)  
 FINCA (Calle, Numero, Municipio, Provincia, Codigo\_postal, Dni\_presidente, Dni\_administrador)

- a. Establecer todas las claves de cada esquema, y proponer una clave primaria para cada uno.
- b. Determinar una cobertura mínima de las dependencias de cada esquema.
- c. ¿Cuál es la forma normal superior de cada esquema?
- d. Normalizar a 3NF los esquemas que no lo sean, siguiendo el algoritmo de normalización 3NF.

- e. Normalizar a BCNF los esquemas que no lo sean (incluyendo los que son 3NF), siguiendo el algoritmo de normalización BCNF. Aplicar también el algoritmo BCNF a los esquemas 3NF obtenidos en el apartado d anterior. Proponer claves primarias para los nuevos esquemas.
- f. Indicar en su caso las dependencias funcionales que se hayan perdido en la normalización.
- En los apartados d y e, mostrar paso a paso la aplicación del algoritmo.

22. En una base de datos de una empresa de inversiones se definen los atributos siguientes:

b: broker	a: acción
o: oficina de un bróker	q: número de acciones de un cliente
c: cliente	d: dividendos pagados por una acción

Sabemos que se cumplen las dependencias funcionales siguientes:

$a \rightarrow d$	$\{c, a\} \rightarrow q$
$c \rightarrow b$	$b \rightarrow o$

Dado un esquema relacional R (b, o, a, q, c, d):

- ¿Dadas las dependencias anteriores, qué claves sabemos que tiene R?
- ¿Está R en segunda forma normal?
- Descomponer R a tercera forma normal.

Supongamos que se descompusiera el esquema R en los dos esquemas siguientes:

R1(c, a, q, d)

R2(c, b, o)

- ¿Están R1 y R2 en tercera forma normal?

23. Dados los siguientes esquemas relacionales:

RESERVA (Nombre\_pasajero, Dni, Nvuelo, Aeropuerto\_origen, Aeropuerto\_destino, Hora, Fecha, Precio)

VUELO (Nvuelo, Hora, Aeropuerto\_origen, Aeropuerto\_destino, Ciudad\_origen, Ciudad\_destino)

Además de las restricciones naturales (p.e. dos personas no pueden tener el mismo DNI), consideramos que se dan las siguientes condiciones:

- Algunas ciudades tienen varios aeropuertos.
- Un mismo vuelo puede tener precios distintos en diferentes reservas.
- No pueden salir dos vuelos con el mismo número el mismo día.
- No pueden salir dos vuelos a la misma hora desde el mismo aeropuerto.
- Un pasajero no puede coger dos veces el mismo vuelo el mismo día, aunque no descartamos que alguien pueda llegar a volar dos veces en distintos vuelos desde el mismo aeropuerto el mismo día (se entiende que es difícil pero no imposible).

Dadas estas condiciones:

- Determinar todas las claves de estos esquemas, y proponer claves primarias.
- Indicar todas las dependencias funcionales que se dan entre los atributos de estas relaciones, evitando redundancias (es decir, si una dependencia se deriva de otras que ya se han indicado, no es necesario indicarla).
- Indicar en qué forma normal se encuentran los esquemas, justificando por qué.
- Convertir ambos esquemas (por separado) a BCNF, indicando las claves primarias de los nuevos esquemas. ¿Se pierde alguna dependencia en la normalización?

24. Demostrar que un esquema de dos atributos siempre es BCNF.

25. Expresar en cálculo relacional las consultas de los ejercicios 2, 5, 6 (excepto 6b) y 7.

26. Expresar en álgebra relacional las consultas de los ejercicios 2, 5, 6 y 7.

27. Dados los siguientes esquemas:

VUELO(Numero,Origen,Destino,Salida)

AEROPUERTO(Codigo,Ciudad)

PASAJERO(Dni,Nombre)

RESERVA(Dni,Numero,Fecha,Precio)

Expresar en cálculo relacional las siguientes consultas:

- Vuelos entre Charles de Gaulle (CDG) y Heathrow (LHR), en esta dirección.
- Hora de salida de los vuelos entre Charles de Gaulle y Heathrow, en esta dirección.
- Vuelos que cubren el trayecto Charles de Gaulle – Heathrow en cualquier sentido.
- Vuelos entre Londres y París (en esta dirección).
- Nombre, Fecha y destino de viaje de todos los pasajeros que vuelan desde Madrid Barajas (MAD).
- Vuelos que no tienen ninguna reserva.

28. Dado el siguiente estado de las relaciones, mostrar el resultado de las consultas del ejercicio anterior.

VUELO				AEROPUERTO	
Numero	Origen	Destino	Salida	Codigo	Ciudad
345	MAD	CDG	12:30	MAD	Madrid
321	MAD	ORY	19:05	LGW	Londres
165	LHR	CDG	09:55	LHR	Londres
903	CDG	LHR	14:40	ORY	París
447	CDG	LHR	17:00	CDG	París

PASAJERO		RESERVA			
Dni	Nombre	Dni	Numero	Fecha	Precio
123	María	789	165	07-01-11	210
456	Pedro	123	345	20-12-10	170
789	Isabel	789	321	15-12-10	250
		456	345	03-11-10	190

29. Sobre la base de datos del ejercicio anterior, mostrar con precisión el resultado de las siguientes consultas:

- $\{p, r \mid \text{PASAJERO}(p) \text{ AND } \text{RESERVA}(r) \text{ AND } r.\text{Precio} < 200\}$
- $\{p, r \mid \text{PASAJERO}(p) \text{ AND } \text{RESERVA}(r) \text{ AND } (r.\text{Dni} = p.\text{Dni} \text{ OR } r.\text{Precio} < 200)\}$
- $\{p, r.\text{Numero} \mid \text{PASAJERO}(p) \text{ AND } \text{RESERVA}(r) \text{ AND } (r.\text{Dni} = p.\text{Dni} \text{ OR } r.\text{Precio} < 200)\}$
- $\{p.\text{Nombre} \mid \text{PASAJERO}(p) \text{ AND } \exists r (\text{RESERVA}(r) \text{ AND } r.\text{Dni} = p.\text{Dni} \text{ AND } r.\text{Precio} < 200)\}$
- $\{p.\text{Nombre} \mid \text{PASAJERO}(p) \text{ AND } \text{NOT } \exists r (\text{RESERVA}(r) \text{ AND } r.\text{Dni} = p.\text{Dni} \text{ AND } r.\text{Precio} < 200)\}$

30. Expresar las consultas de los ejercicios 27 y 29 en álgebra relacional.

31. Escribir en SQL las consultas de los ejercicios 27 y 29.

32. Mostrar el resultado de las siguientes consultas sobre el estado de datos del ejercicio 28:

- $\pi_{\text{Dni}} (\sigma_{\text{Precio} > 200} (\text{RESERVA}))$
- $\sigma_{\text{Nombre} = \text{'María'}} (\text{PASAJERO} \times \text{RESERVA})$
- $\sigma_{\text{Nombre} = \text{'María'}} (\text{PASAJERO} \bowtie \text{RESERVA})$
- $\sigma_{\text{Dni} = 123} (\text{PASAJERO} \bowtie \text{RESERVA})$
- $\pi_{\text{Nombre, Precio}} (\sigma_{\text{Nombre} = \text{'Pedro'}} (\text{PASAJERO} \bowtie \text{RESERVA})) \cap \pi_{\text{Nombre, Precio}} (\sigma_{\text{Precio} > 100} (\text{PASAJERO} \times \text{RESERVA}))$
- $R1 \leftarrow \rho_{\text{Ciudad, Ciudad\_origen}} (\sigma_{\text{Ciudad} = \text{'Madrid'}} (\text{VUELO} \bowtie_{\text{Origen, Codigo}} \text{AEROPUERTO}))$   
 $R2 \leftarrow \rho_{\text{Ciudad, Ciudad\_destino}} (\sigma_{\text{Ciudad} = \text{'Londres'}} (\text{VUELO} \bowtie_{\text{Destino, Codigo}} \text{AEROPUERTO}))$   
 $\pi_{\text{Numero}} (R1 \bowtie R2)$

33. ¿Cabe reescribir alguna de las consultas del ejercicio anterior para optimizar el coste de su ejecución? En tal caso, mostrar y explicar la reformulación.

34. Expresar las consultas del ejercicio 32 en cálculo relacional.

35. Escribir en SQL las consultas del ejercicio 32.

36. Dados los siguientes esquemas y datos:

PELICULA

Titulo	Genero
Annie Hall	Comedia
Espartaco	Aventura
Fargo	Drama
Manhattan	Comedia

OSCAR

Año	Categoria	Premiado	Pelicula
1961	Actor	P. Ustinov	Espartaco
1977	Director	W. Allen	Annie Hall
1997	Guión	J. Cohen	Fargo

DIRECCION

Película	Director
Annie Hall	W. Allen
Espartaco	S. Kubrick
Fargo	J. Cohen
Manhattan	W. Allen

GUION

Pelicula	Guionista
Annie Hall	W. Allen
Espartaco	H. Fast
Fargo	J. Cohen
Manhattan	W. Allen

REPARTO

Pelicula	Actor	Papel
Annie Hall	W. Allen	Alvy Singer
Annie Hall	D. Keaton	Annie Hall
Espartaco	K. Douglas	Espartaco
Espartaco	P. Ustinov	Batiatus
Fargo	F. McDormand	Marge
Manhattan	W. Allen	Isaac
Manhattan	D. Keaton	Mary

- a. Formular en cálculo relacional las siguientes consultas:
  - i. ¿Quién dirigió Espartaco?
  - ii. ¿Qué actores han sido compañeros de reparto de P. Ustinov?
  - iii. ¿Qué cineastas han sido a la vez directores, guionistas e intérpretes en una misma película?
  - iv. ¿Qué actores no han trabajado en películas dirigidas por W. Allen?
- b. Mostrar el resultado de las siguientes consultas:
  - i.  $\{ r.Actor \mid \text{REPARTO}(r) \text{ AND } \exists p (\text{PELICULA}(p) \text{ AND } p.Genero = 'Aventura' \text{ AND } r.Pelicula = p.Pelicula) \}$
  - ii.  $\{ g.Pelicula \mid \text{GUION}(g) \text{ AND } \exists o (\text{OSCAR}(o) \text{ AND } o.Premiado = g.Guionista) \}$
  - iii.  $\{ d.Director, r.Actor \mid \text{DIRECCION}(d) \text{ AND } \text{REPARTO}(r) \text{ AND } d.Pelicula = r.Pelicula \text{ AND NOT } \exists o (\text{OSCAR}(o) \text{ AND } o.Pelicula = d.Pelicula) \}$
  - iv.  $\{ p.Titulo, r.Papel \mid \text{PELICULA}(p) \text{ AND } p.Genero \neq 'Comedia' \text{ AND } \text{REPARTO}(r) \text{ AND NOT } \exists o (\text{OSCAR}(o) \text{ AND } o.Pelicula = r.Pelicula) \}$

37. ¿Cuál es el resultado de las siguientes expresiones de álgebra relacional sobre las relaciones del ejercicio anterior?

- a.  $\pi_{Premiado} (\sigma_{Año > 1970} (\text{OSCAR}))$
- b.  $\pi_{Director, Actor} (\sigma_{Pelicula = 'Manhattan'} (\text{DIRECCION} \bowtie \text{REPARTO}))$

38. Escribir en álgebra relacional las consultas del ejercicio 36.

39. Escribir en SQL las consultas del ejercicio 36.

40. Escribir en SQL las consultas del ejercicio 37.

41. Un sitio web de música almacena información sobre discos, canciones, músicos y usuarios. Los usuarios pueden seguirse unos a otros formando una red social asimétrica (tipo Twitter). Cada vez que un usuario escucha una canción, el acceso a la misma queda registrado en el sistema. Toda esta información se almacena en una base de datos.

Suponiendo que el estado de la base de datos es el siguiente:

DISCO		MUSICO		CANCION			
ID	Titulo	ID	Nombre	Disco	Pista	Titulo	Interprete
102	Échate un cantecito	555	Kiko Veneno	102	3	Echo de menos	555
200	Mosaïque	632	John Lennon	200	1	Caminando por la calle	830
460	Imagine	830	Gipsy Kings	200	3	Mosaïque	830
705	Compás			460	9	Julia	632
				460	16	Stand by me	632
				705	11	Di me	830

USUARIO		SIGUE		ESCUCHA			
Nick	Nombre	Seguidor	Usuario	Disco	Pista	Usuario	Instante
charo	Rosario	charo	paco	102	3	maite	2010-11-29 15:07:35
fede	Federico	charo	maite	200	1	paco	2010-12-15 22:10:48
maite	María Teresa	fede	nacho	200	3	nacho	2011-01-02 11:05:44
nacho	Ignacio	maite	nacho	200	3	nacho	2011-01-02 11:40:02
paco	Francisco	nacho	maite	460	9	fede	2010-12-15 22:10:48
		nacho	charo	460	16	fede	2011-01-04 12:35:13
				705	11	maite	2011-01-01 20:14:27

- Escribir en cálculo relacional las siguientes consultas:
  - Nombre de los seguidores de 'maite'.
  - Seguidores de 'nacho' que no han escuchado ninguna canción titulada 'Stand by me'.
- Escribir en álgebra relacional las siguientes consultas:
  - Título de las canciones que ha escuchado 'nacho'.
  - Nombre de los usuarios que han escuchado alguna canción de 'John Lennon'.
- Mostrar (con exactitud formal) el resultado de las siguientes consultas:
  - $$\{ u, c.Titulo \mid \text{USUARIO}(u) \text{ and } \text{CANCION}(c) \\ \text{and } \exists e (\text{ESCUCHA}(e) \text{ and } e.Disco = c.Disco \text{ and } e.Pista = c.Pista \text{ and } e.Usuario = u.Nick) \\ \text{and } \exists m (\text{MUSICO}(m) \text{ and } m.ID = c.Interprete \text{ and } m.Nombre = 'John Lennon') \}$$
  - $$S \leftarrow \pi_{\text{Usuario}} (\text{SIGUE} \bowtie_{\text{Seguidor} = \text{Nick}} (\sigma_{\text{Nombre} = 'Rosario'} (\text{USUARIO}))) \\ \pi_{\text{Nombre, ID}} (\text{MUSICO} \bowtie_{\text{ID} = \text{Interprete}} (\text{CANCION} \bowtie (S \bowtie \text{ESCUCHA})))$$

Nota: en los apartados a) y b), al escribir consultas se entiende que uno no conoce de antemano los datos de la BD.

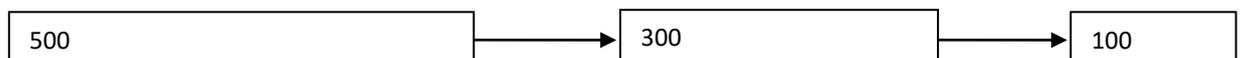
42. Escribir en SQL las consultas del ejercicio anterior.

43. Dada la relación RESERVA del ejercicio 28:

- ¿Tendría sentido almacenar los datos de la relación en un fichero con registros de longitud variable? Explicar la respuesta.
- Suponiendo que los datos de la relación RESERVA se almacenan con registros de longitud fija:
  - Cancelar la reserva de María.
  - Cancelar la reserva de Pedro.
  - Reservar un billete para María en el vuelo 321 el 23-12-10, por 280 euros.

Después de cada operación, mostrar con detalle el estado del almacenamiento de la tabla de registros, con las estructuras necesarias para gestionar la eliminación y la inserción por medio de una lista de registros borrados.

44. En un fichero con registros de longitud variable, el borrado de registros se gestiona con estrategia *worst fit*. En un momento dado, la lista de registros borrados es la siguiente (los números indican la longitud de cada registro):



Mostrar cómo queda la lista de registros borrados tras cada una de las siguientes operaciones:

- Se inserta un registro de 300 bytes.

- b. Se elimina un registro de 250 bytes.
- c. Se inserta un registro de 400 bytes.

Nota: aplicar cada operación sobre el resultado de la operación anterior.

45. La siguiente tabla representa (de forma simplificada) el registro de trabajos en curso en un taller de mecánica (las celdas vacías corresponden a valores NULL):

TRABAJO			
Matricula	Descripcion	Fecha	Observaciones
0123ABC	Chapa	12-11-10	Mapfre
4567DEF	Revisión	12-11-10	Coche de sustitución
8901GHI	Cambio de aceite	15-11-10	
2345JKL	Substitución neumáticos delanteros	15-11-10	Pirelli P6 four seasons
6789MNO	Paralelo y equilibrado	15-11-10	

- a. Mostrar con detalle cómo se almacenarían los datos en registros de longitud variable con indicadores de longitud.
  - b. Mostrar con detalle el estado del fichero después de aplicar cada una de las siguientes operaciones, utilizando la estrategia first-fit:
    - i. Se finalizan los trabajos con los vehículos 0123ABC, 2345JKL y 8901GHI (por este orden), y se eliminan por tanto los registros correspondientes.
    - ii. El 16-11-10 entran al taller dos vehículos: uno con matrícula 2222PQR para hacer una "Revisión", sin observaciones; y poco después otro con matrícula 5555STU, para una "Reparación elevallunas", también sin observaciones. Insertar los registros correspondientes, en este orden.
  - c. Repetir el apartado anterior utilizando la estrategia best-fit.
  - d. Repetir el apartado con la estrategia worst-fit.
46. Estimar el coste en tiempo necesario para leer 5.000.000 registros de 400B cada uno, en un dispositivo de disco con las siguientes características:
- Tiempo de seek promedio: 8ms.
  - Velocidad de rotación: 15.000 rpm.
  - Tamaño de sector: 1.000B.
  - Nº promedio de sectores por pista: 500.
  - Tamaño de bloque: 4 sectores.
- Realizar la estimación suponiendo tres casos diferentes:
- a. Los registros se leen con una instrucción del programa que lee los registros de uno en uno.
  - b. Los registros se leen sobre un buffer de 400.000B.
  - c. El fichero se lee entero en una sola instrucción del programa.
- En los tres apartados, considerar dos situaciones diferentes:
- i. El fichero que contiene los registros está almacenado en bloques contiguos (mejor caso).
  - ii. El fichero está totalmente fragmentado en bloques separados (peor caso).
47. Mostrar con detalle las estructuras necesarias para crear los siguientes índices simples:
- a. Para los datos del ejercicio 28:
    - i. Un índice primario para las tablas VUELO y RESERVA.
    - ii. Índices secundarios por Origen y Destino en la tabla VUELO.
    - iii. Un índice secundario por Dni en la tabla reserva.
  - b. Para la tabla TRABAJO del ejercicio 45:
    - i. Un índice primario.
    - ii. Un índice secundario por fecha.
- Optar razonadamente en cada caso entre una implementación con registros de longitud variable o fija.

48. Utilizando los índices del ejercicio anterior, explicar y mostrar cómo se ejecutaría una búsqueda de vuelos entre Charles de Gaulle (CDG) y Heathrow (LHR) utilizando los índices correspondientes.
49. Explicar qué actualizaciones son necesarias en los índices del ejercicio 47 al realizar las siguientes operaciones:
- Las eliminaciones e inserciones del apartado b del ejercicio 45.
  - Cancelar todas las reservas de Isabel (sabiendo su Dni) en la tabla RESERVA.
- Mostrar el estado de los índices después de cada operación.
50. En la base de datos del ejercicio 41, y suponiendo que los datos se almacenan en registros de longitud fija, mostrar con detalle las estructuras necesarias para crear:
- Un índice primario de la tabla CACION (se entiende que una canción –su título– se puede reeditar en discos diferentes).
  - Un índice por intérprete de la misma tabla.
  - Un índice por disco de la misma tabla.

Repetir el ejercicio suponiendo que ahora:

- Los IDs de discos e intérpretes se guardan con tipo long (4 bytes) en binario.
- Los nºs de pista se guardan en binario con 1 byte.
- Los títulos de canciones se guardan en modo texto con longitud variable, utilizando un contador de longitud de tipo short (2 bytes) en binario.
- La longitud de los registros se indica con un número de 2 bytes.

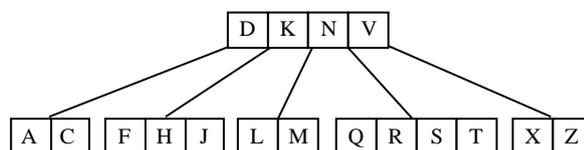
51. En el ejercicio anterior, sobre la opción de longitud variable:
- Modificar el título “Mosaïque” por “Luna de fuego”.
  - Eliminar el registro “Caminando por la calle”.
  - Eliminar el registro “Stand by me”.
  - Modificar “Julia” por “Imagine”.

En los apartados a - d, mostrar con detalle el estado del fichero de datos utilizando las siguientes estrategias:

- First fit.
- Best fit.
- Worst fit.

52. Mostrar el estado de los índices del ejercicio 50 al final de todas las operaciones del ejercicio 51.

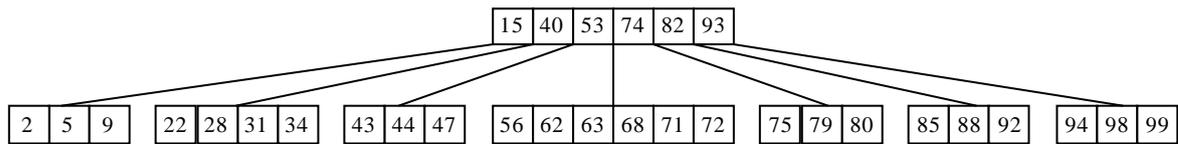
53. Dado el siguiente árbol B, con páginas de tamaño para 4 claves:



- Insertar la clave P.
- Partiendo del resultado anterior, eliminar L y J.

Mostrar el estado del árbol después de cada operación.

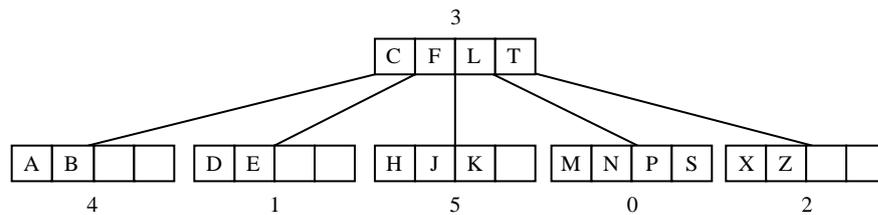
54. Dado el siguiente árbol B de orden 6:



- Insertar la clave 60.
- En el árbol resultante, eliminar 2 y 63.

Mostrar el estado del árbol después de cada operación.

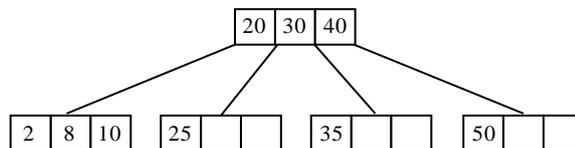
55. Dado el siguiente árbol B, con páginas para 4 claves:



- Insertar la clave R.
- Sobre el árbol resultante, eliminar la clave B.
- Mostrar cómo se almacenaría en un fichero el árbol inicial.

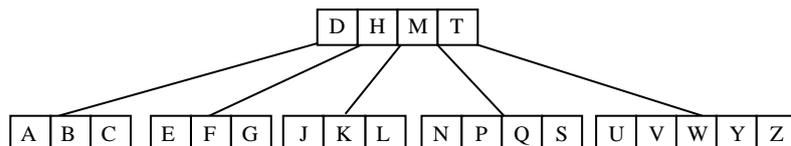
Mostrar el estado del árbol después de cada operación.

56. En el siguiente árbol B:



- Insertar la clave 5.
- En el árbol resultante, eliminar la clave 40.
- En el árbol resultante, eliminar las claves 50 y 35.
- Estimar el coste en tiempo de la operación del apartado a) suponiendo lo siguiente:
  - Utilizamos un disco con 100 bloques por pista, seek promedio de 3ms y rotación promedio de 2ms.
  - Las páginas tienen tamaño menor que los bloques. Al principio de la operación no hay ninguna página en RAM, pero hay espacio suficiente para mantener hasta 5 páginas en RAM simultáneamente.

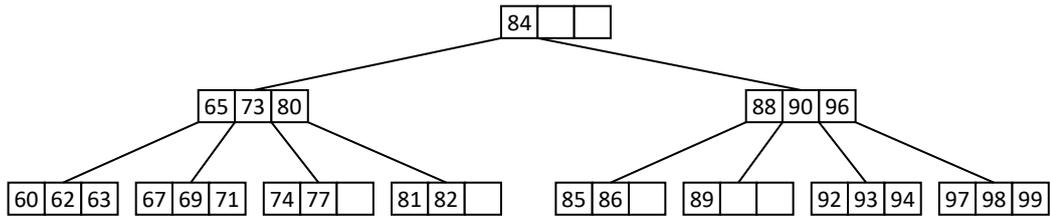
57. Dado el siguiente árbol B\*, con páginas para 5 claves:



- Insertar X.
- Partiendo del resultado anterior, insertar R.

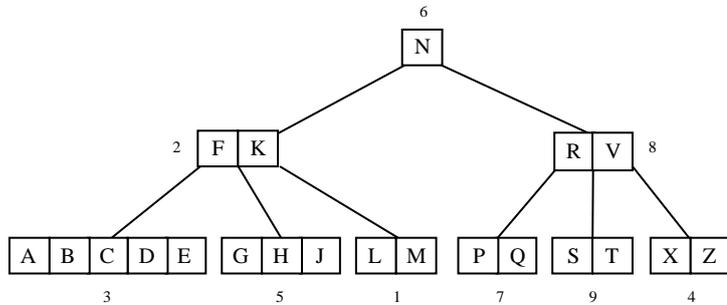
Mostrar el estado del árbol después de cada operación

58. En el siguiente árbol B de 3 claves por página:



- Insertar la clave 40.
- En el árbol resultante, insertar la clave 101.
- Volviendo al estado inicial del árbol y considerándolo como un árbol B\* de 3 claves por página, insertar la clave 70.
- En el árbol resultante ahora, insertar la clave 40.
- Demostrar que en general la altura,  $h$ , de un árbol B viene acotada por  $h \leq 1 + \log_{m+1} \frac{n+1}{2}$ , donde  $m$  es el mínimo número de claves por página y  $n$  es el número de claves contenidas en el árbol.

59. Muestra con detalle cómo se almacenaría en un fichero el siguiente árbol B (con páginas para 5 claves), donde se indica el número de página junto a cada una.



60. En un fichero se almacenan los datos de un árbol B con el siguiente contenido (el campo "head" en la cabecera es un puntero a la lista de páginas borradas):

Raíz	3						
Head	-1						
0	1	R		6	4		
1	3	A	C	D	-1	-1	-1
2	3	G	H	J	-1	-1	-1
3	1	N		5	0		
4	1	S		-1	-1		
5	1	E		1	2		
6	1	P		-1	-1		

- Dibujar la estructura que tendría el árbol B cuyos datos son los que se almacenan en el fichero.
- Mostrar cómo queda el fichero cuando:
  - Se inserta la clave F.
  - Se elimina la clave R.
  - Se elimina la clave P.
  - Se inserta la clave B.

Efectuar cada operación sobre el resultado de la anterior, y utilizar una lista de páginas borradas para gestionar la eliminación e inserción de páginas. Mostrar el resultado después de cada operación.

- Un árbol B de 3 claves por página contiene las claves: A, B, C, D, E, F, G, H, I, J, K, L, M, N, O.
  - Dibujar el árbol óptimo (número mínimo de páginas) para almacenar estas claves.
  - Insertar la clave P en este árbol.
  - Dibujar el árbol en el peor caso (máximo coste en la búsqueda).
  - Eliminar la clave L de este árbol.

62. Con las mismas claves del ejercicio anterior, dibujar un árbol B+ óptimo de 3 claves por página.
63. Partiendo del índice primario de la tabla VUELO creado en el ejercicio 47.a.i:
- Crear el mejor árbol B posible para este índice con páginas de dos claves.
  - Crear el mejor árbol B+ posible para este índice con páginas de dos claves.
  - Mostrar el peor árbol B y B+ posible con páginas de dos claves.
  - Mostrar con todo detalle cómo se almacenarían en disco los árboles B y B+ de los apartados a y b (incluyendo las referencias a las posiciones de registros).
64. Partiendo del árbol B+ de los apartados b y c del ejercicio anterior:
- Insertar un nuevo vuelo con nº 201 en el fichero de registros.
  - A continuación, eliminar los vuelos 903 y 447.
- Mostrar cómo queda el árbol B+ después de cada operación, tanto en forma de diagrama como en las estructuras en disco.
65. Dado un árbol B con páginas de tamaño para  $k$  claves:
- Si la altura del árbol es  $h$  y todas las páginas están completamente llenas, ¿cuántas claves hay en total en el árbol?
  - ¿Cómo cambia la respuesta al apartado anterior si en lugar de ser un árbol B es un árbol B+?

66. Tenemos un campo de una tabla indexado mediante hashing extensible. Utilizamos bloques con capacidad para dos claves, y una función hash que devuelve los siguientes valores para las claves siguientes:  $h(a) = 3$ ,  $h(b) = 9$ ,  $h(c) = 1$ ,  $h(d) = 2$ ,  $h(e) = 7$ ,  $h(f) = 5$ ,  $h(g) = 4$ ,  $h(j) = 10$ ,  $h(k) = 13$ . Mostrar paso a paso el estado de las estructuras de este método de indexación (directorio y bloques de claves) cuando se realizan las siguientes operaciones.

- a. Partiendo "de cero", insertar las claves  $a, b, c, d, e, f, g, j$ , por este orden.
- b. Partiendo del resultado del apartado anterior, eliminar  $f, b$  y  $a$ .
- c. Partiendo de cero de nuevo, insertar las claves  $c, b$  y  $f$ .