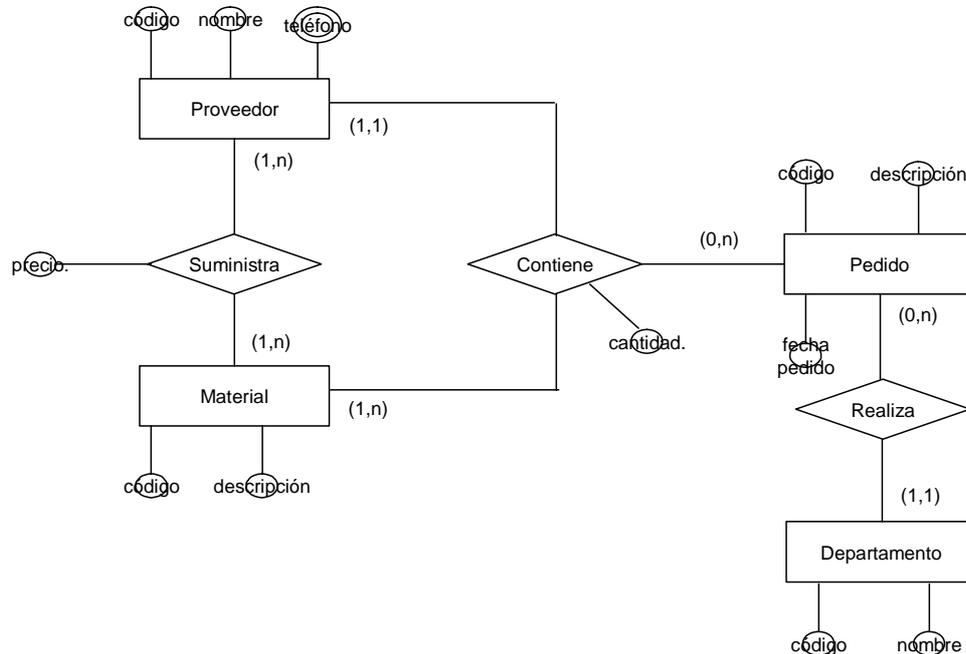


Examen de Bases de datos y sistemas de información. Final de Junio (I PARCIAL)

1) (3,6 puntos) Dado el siguiente modelo E/R, realizar:



a.) Convertir el modelo E/R anterior en modelo relacional en 3FN expresando las restricciones de integridad referencial en notación algebraica.

```
Proveedor(#código, nombre)
Teléfono(#código, teléfono)
Material(#código, descripción)
Pedido(#código, descripción, fecha_pedido, código_departamento)
Departamento(#código, nombre)
Suministra(#código_proveedor, #código_material, precio)
Contiene(#código_proveedor, #código_material, #código_pedido, cantidad)
```

Restricciones de integridad referencial:

$$\prod_{código} (Teléfono) \subseteq \prod_{código} (Proveedor)$$
$$\prod_{código_departamento} (Pedido) \subseteq \prod_{código} (Departamento)$$
$$\prod_{código_proveedor} (Suministra) \subseteq \prod_{código} (Proveedor)$$
$$\prod_{código_material} (Suministra) \subseteq \prod_{código} (Material)$$
$$\prod_{código_proveedor} (Contiene) \subseteq \prod_{código} (Proveedor)$$
$$\prod_{código_material} (Contiene) \subseteq \prod_{código} (Material)$$
$$\prod_{código_pedido} (Contiene) \subseteq \prod_{código} (Pedido)$$

b.) Realizar en SQL las siguientes consultas a partir del modelo relacional del punto anterior:

i. Un listado con todos los departamentos y los productos que han comprado (si los hay), indicando el código y nombre del departamento, el código de material, la descripción, la cantidad y la fecha del pedido.

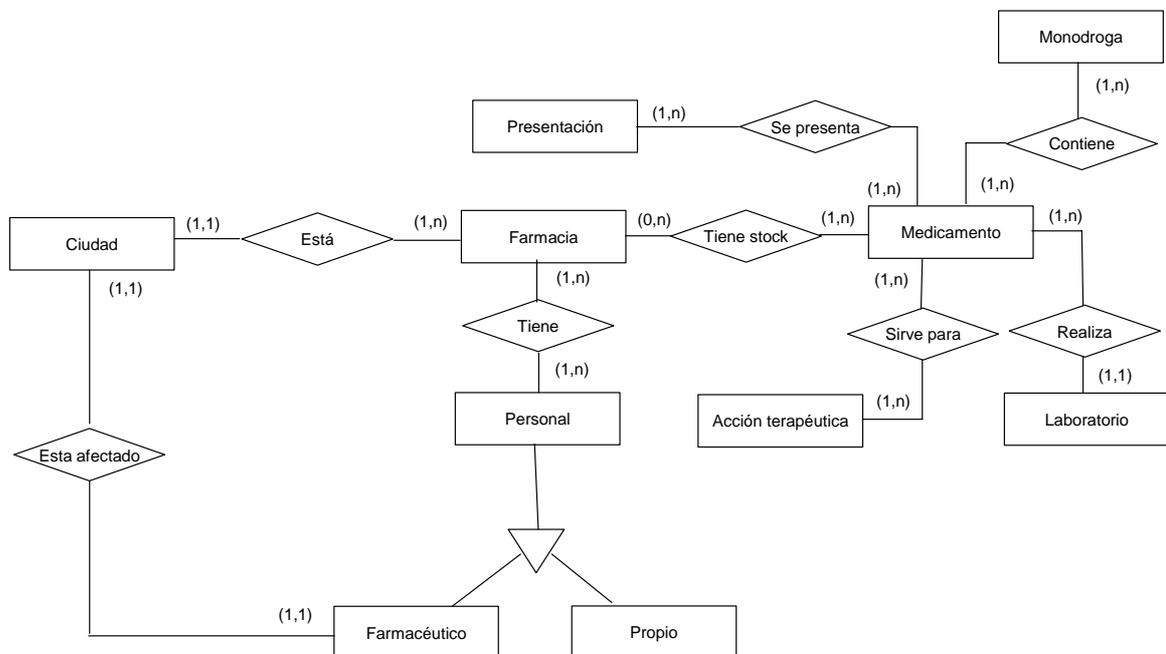
```
select Departamento.código, Departamento.nombre, Material.código,
Material.descripcion, Contiene.cantidad, Pedido.fecha_pedido
from Departamento
left outer join Pedido on Departamento.codigo = Pedido.codigo_departamento inner
join Contiene on Pedido.codigo = Contiene.codigo_pedido
inner join Material on Contiene.codigo_material = Material.codigo;
```

ii. Un listado con todos los proveedores y los materiales que suministran con la cantidad total suministrada.



```
select Contiene.código_proveedor, Proveedor.nombre, Contiene.código_material,
Material.descripcion, SUM(Contiene.cantidad)
from Contiene, Proveedor, Material
where Contiene.código_proveedor = Contiene.código
and Contiene.código_material = Material.codigo
group by Contiene.código_material;
```

- 2) (3,3 puntos) Se desea mantener una base de datos para una cadena de farmacias distribuida en diferentes ciudades. Cada farmacia tiene sus empleados propios y un farmacéutico. Por cada ciudad existe un único farmacéutico; esto es, si en una ciudad hubiera más de una farmacia, el mismo farmacéutico estaría afectado a todas las farmacias de esa ciudad. Cada farmacia tiene a su vez su stock de medicamentos. El mismo se mantiene por medicamento y presentación. Los medicamentos se organizan según la o las monodrogas que lo componen, su presentación (por ejemplo ampollas de 5 unidades, jarabe de 100ml, inyecciones por 10 unidades, pomada 60gr, etc.), el laboratorio que lo comercializa, y su acción terapéutica (analgésico, antibiótico, etc.). Por cada medicamento se mantiene su precio y la cantidad en existencia del mismo. El sistema deberá permitir consultar la base de datos de diferentes alternativas para medicamentos compuestos por una monodroga, medicamentos de un laboratorio, presentaciones de un medicamento, entre otras. Se pide:
- Construir el esquema entidad/relación añadiendo el mínimo número de atributos, imponiendo y explicando las restricciones de cardinalidad mínimo-máximo, sin considerar los costes de las operaciones pero asegurando que se puedan realizar.



- Traducir el esquema conceptual a un esquema lógico relacional.

```
Ciudad (#nombre, dni_farmacéutico)
Farmacia (#número, nombre, dirección, nombre_ciudad)
Personal (#dni, nombre)
Farmacéutico (#dni, fecha_título, nombre_ciudad)
Propio (#dni, salario)
Medicamento (#código, nombre, precio, cantidad, código_laboratorio)
Acción terapéutica (#tipo_acción, descripción)
Laboratorio (#código, nombre)
Presentación (#tipo_presentación)
Monodroga (#código, descripción)
Tiene (#número_farmacia, #dni_personal)
Tiene_stock (#número_farmacia, #código_medicamento)
Se_presenta (#código_medicamento, #tipo_presentación)
Contiene (#código_medicamento, #código_monodroga)
Sirve_para (#código_medicamento, #tipo_acción)
```

- 3) (0,6 puntos) Dada la siguiente tabla R(ABCDEG) y el siguiente conjunto de dependencias funcionales:



$\{BC \rightarrow A, A \rightarrow EG, A \rightarrow D, EG \rightarrow D, D \rightarrow A\}$

Se pide calcular todas las claves. Fundamente su respuesta.

Los atributos BC no son determinados por ningún otro atributo, por lo que deben participar en todas las claves.

$(BC)^+ = \{B, C, A, E, G, D\}$ por lo que BC es clave. Como estos atributos deben participar en todas las claves, entonces es la única clave.

4) (1,25 puntos) Dadas las siguientes tablas, traducir a QBE la siguiente consulta SQL:

Persona (#DNI, nombre, dirección)

Coche (#matrícula, año, modelo)

Dueño (#DNI, #matrícula)

```
select Persona.nombre, Persona.dirección, Coche.matrícula, Coche.modelo
```

```
from Persona
```

```
natural inner join Dueño
```

```
natural inner join Coche;
```

Persona	DNI	nombre	dirección
	_x	P	P

Coche	matrícula	año	modelo
	P._y		P

Dueño	DNI	matrícula
	_x	_y

5) (1,25 puntos) Dados los siguientes esquemas relacionales $R = (a, b)$ y $S = (a, c)$, escribir en álgebra relacional y en cálculo relacional de tuplas las siguientes expresiones del cálculo relacional de dominios:

1. $\{ \langle a \rangle \mid \exists b (\langle a, b \rangle \in R \wedge b = 17) \}$

AR: $\pi_a(\sigma_{b=17}(R))$

CRT: $\{ t \mid \exists s \in R (t[a] = s[a] \wedge s[b] = 17) \}$

2. $\{ \langle a, b, c \rangle \mid \langle a, b \rangle \in R \wedge \langle a, c \rangle \in S \}$

AR: $R \times S$

CRT: $\{ t \mid \exists r \in R (\exists s \in S (t[a] = r[a] \wedge r[a] = s[a] \wedge t[b] = r[b] \wedge t[c] = s[c])) \}$



Examen de Bases de datos y sistemas de información. Final de Junio (II PARCIAL)

- 6) (1,9 puntos) Considérese la siguiente planificación concurrente de las transacciones T y U que acceden a las cuentas bancarias A, B y C:

T	U
saldo = B.leer();	saldo = B.leer();
B.escribir(saldo*1.1);	B.escribir(saldo*1.1);
A.restar(saldo/10);	C.restar(saldo/10);

Con los valores iniciales de las cuentas: A = 100€ B = 200€y C = 300€ se pide:

- a. Dar los valores de las cuentas tras la ejecución de las transacciones de acuerdo a la planificación dada.

T	U
saldo = B.leer(); 200€	saldo = B.leer(); 200€
B.escribir(saldo*1.1); 220€	B.escribir(saldo*1.1); 220€
A.restar(saldo/10); 80€	C.restar(saldo/10); 280€

A = 80€ B = 220€ C = 280€

- b. Dar los valores de las cuentas si la ejecución de las transacciones hubiera sido secuencial, empezando por la transacción T. Si los resultados son diferentes a los del punto anterior, explicar razonadamente las diferencias.

A = 80€ B = 242€ C = 278€

La diferencia se produce porque en la planificación se produce el efecto de modificación perdida (“lost update”), ya que la actualización de la cuenta B que realiza la transacción U se pierde.

- c. Dar una planificación no secuencial que sea secuenciable.

T	U
saldo = B.leer(); 200€	saldo = B.leer(); 220€
B.escribir(saldo*1.1); 220€	B.escribir(saldo*1.1); 242€
A.restar(saldo/10); 80€	C.restar(saldo/10); 278€

- 7) (1,2 puntos) Dado el siguiente esquema:

Centro_estudios (#cod_centro, dirección, teléfono, num_alumnos)

Alumno (#cod_alumno, nombre, dirección, cod_centro)

$\prod_{cod_centro} (Alumno) \subseteq \prod_{cod_centro} (Centro_estudios)$

Se pide diseñar y especificar en Oracle los disparadores que se estimen oportunos para mantener actualizado el número de alumnos matriculados en cada centro de estudios toda vez que se altere la tabla Alumno.

```
CREATE OR REPLACE TRIGGER actualizar_alumno
AFTER INSERT OR UPDATE OR DELETE ON ALUMNO
FOR EACH ROW
```



```
BEGIN
  IF DELETING THEN
    UPDATE centro_estudios
    SET centro_estudios.num_alumnos= centro_estudios.num_alumnos-1
    WHERE centro_estudios.cod_centro=:OLD.cod_centro;
  ELSEIF INSERTING
    UPDATE centro_estudios
    SET centro_estudios.num_alumnos= centro_estudios.num_alumnos+1
    WHERE centro_estudios.cod_centro=:OLD.cod_centro;
  ELSE
    UPDATE centro_estudios
    SET centro_estudios.num_alumnos= centro_estudios.num_alumnos-1
    WHERE centro_estudios.cod_centro=:OLD.cod_centro;
    UPDATE centro_estudios
    SET centro_estudios.num_alumnos= centro_estudios.num_alumnos+1
    WHERE centro_estudios.cod_centro=:NEW.cod_centro;
  END IF;
END;
```

8) (1,9 puntos) Dado el siguiente esquema relacional:

Programas(nomProg, genero, actorPpal)

Programación(nomProg, dia, horaComienzo, horaFin)

Empleados(nomEmp, ciEmp, dirEmp, telEmp, genero, cargo, horaEntrada, horaSalida)

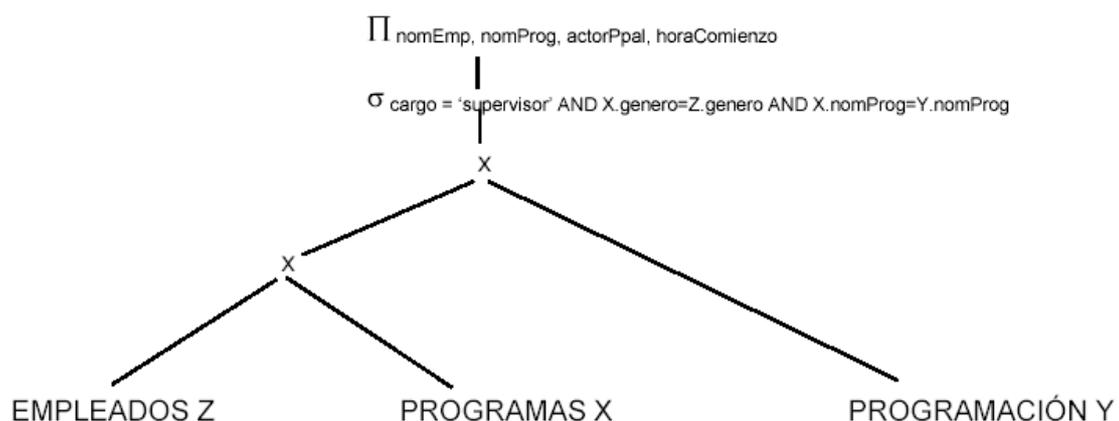
y la siguiente consulta:

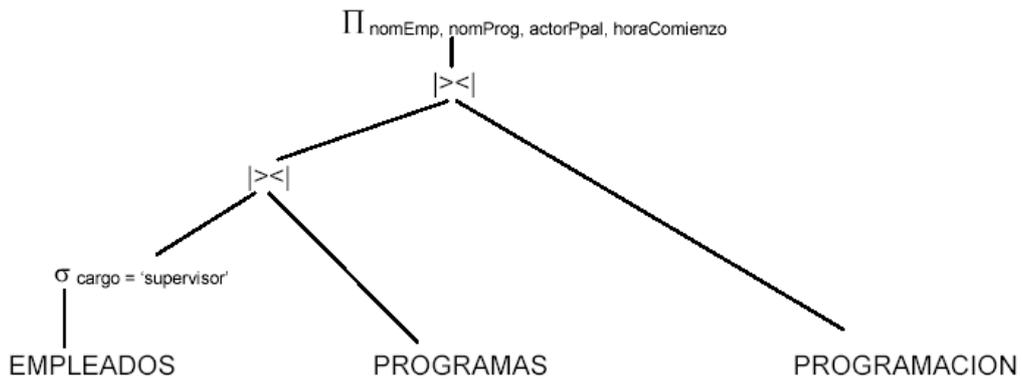
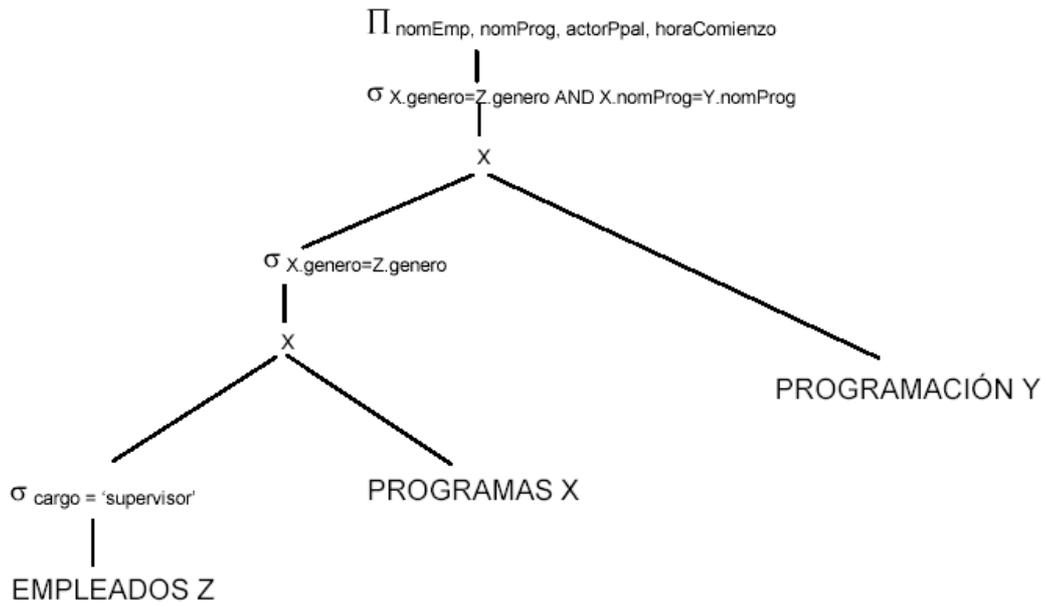
```
SELECT nomEmp, nomProg, actorPpal, horaComienzo
```

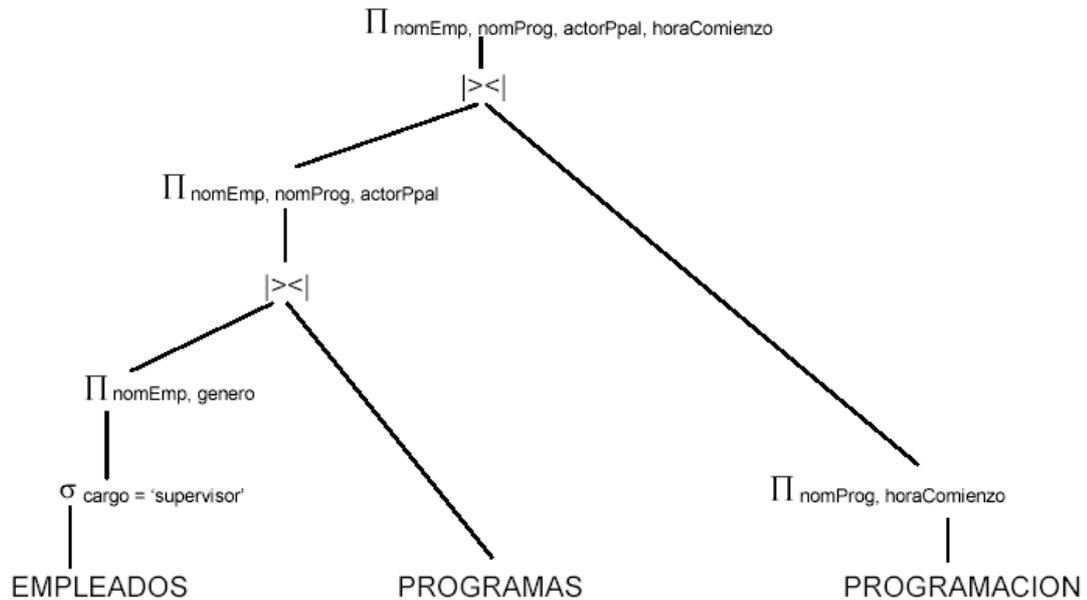
```
FROM Programas X, Programación Y, Empleados Z
```

```
WHERE X.genero = Z.genero AND X.nomProg = Y.nomProg AND Z.cargo = 'supervisor'
```

Dar un plan lógico para la consulta aplicando las heurísticas para obtenerlo.







9) (2,5 puntos) Dada la secuencia ordenada de valores de clave se pide para $n=4$ (número de punteros):

a. Construir un árbol B+ insertando valores en el orden de la secuencia.

Restricciones:

Nodo raíz: entre 1 y 4 hijos (1 y n).

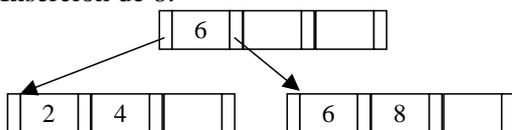
Nodo hoja: entre 2 y 3 valores ($\lceil (n-1)/2 \rceil$ y n).

Nodos internos: entre 2 y 4 hijos ($\lceil n/2 \rceil$ y n).

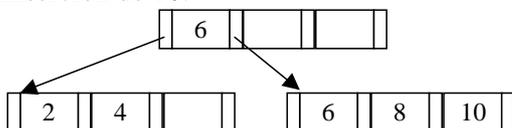
Inserción de 2,4,6 en el nodo raíz:



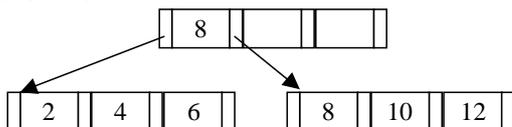
Inserción de 8:



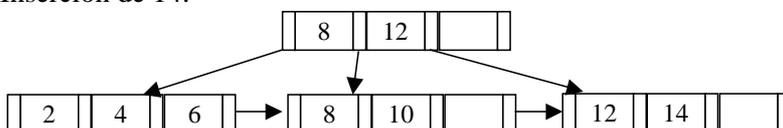
Inserción de 10:



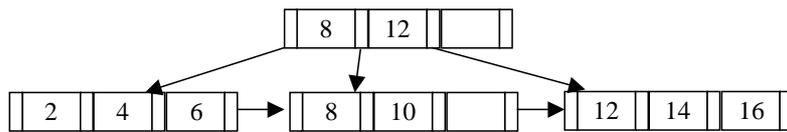
Inserción de 12:



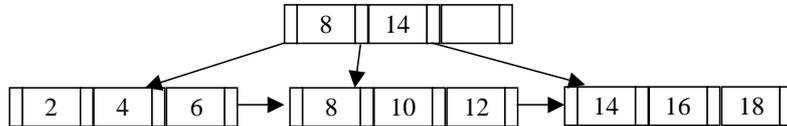
Inserción de 14:



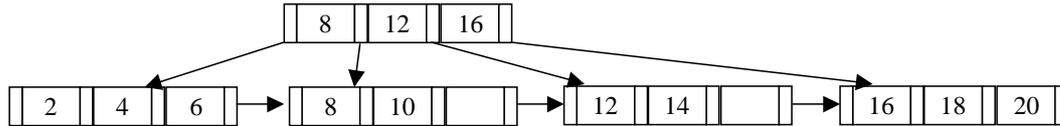
Inserción de 16:



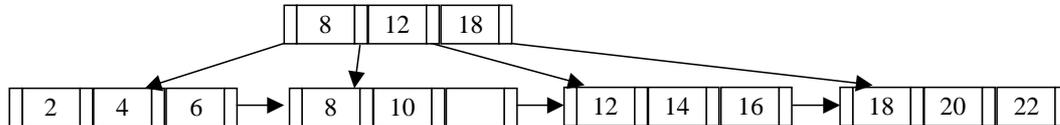
Inserción de 18:



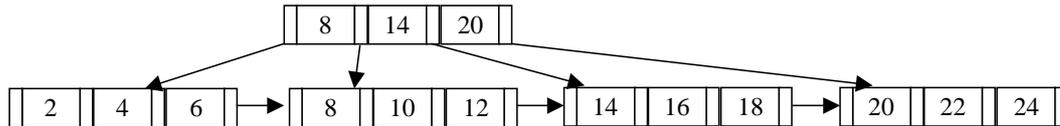
Inserción de 20:



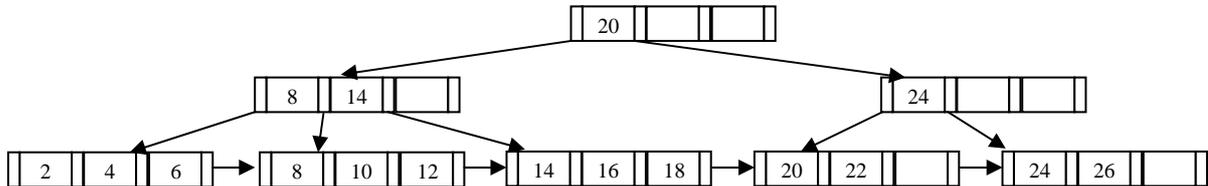
Inserción de 22:



Inserción de 24:



Inserción de 26:



b. Construir un árbol B insertando valores en el orden de la secuencia.

Restricciones:

Nodo raíz: entre 1 y 4 hijos (1 y n).

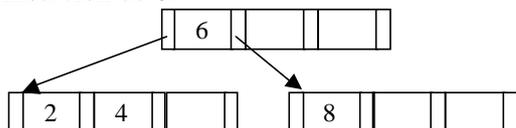
Nodo hoja: entre 1 y 3 valores ($\lceil n/2 \rceil - 1$ y n).

Nodos internos: entre 2 y 4 hijos ($\lceil n/2 \rceil$ y n).

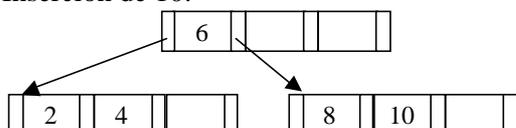
Inserción de 2,4,6 en el nodo raíz:



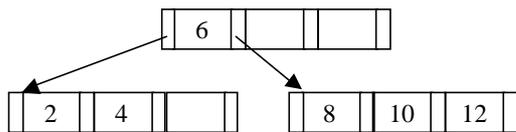
Inserción de 8:



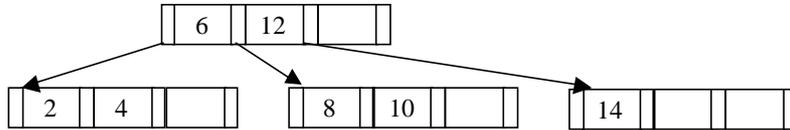
Inserción de 10:



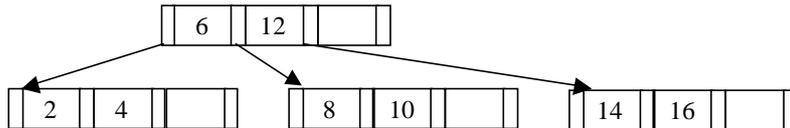
Inserción de 12:



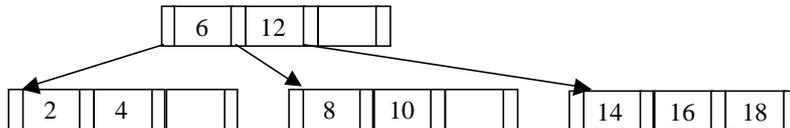
Inserción de 14:



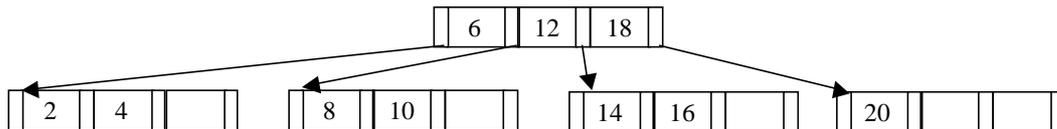
Inserción de 16:



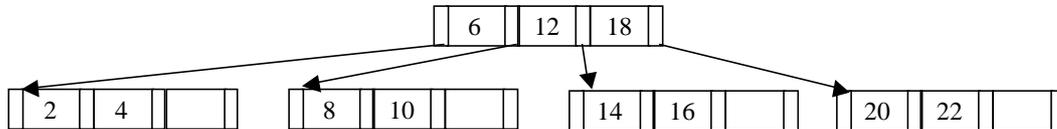
Inserción de 18:



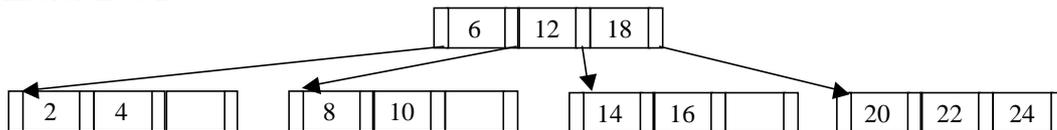
Inserción de 20:



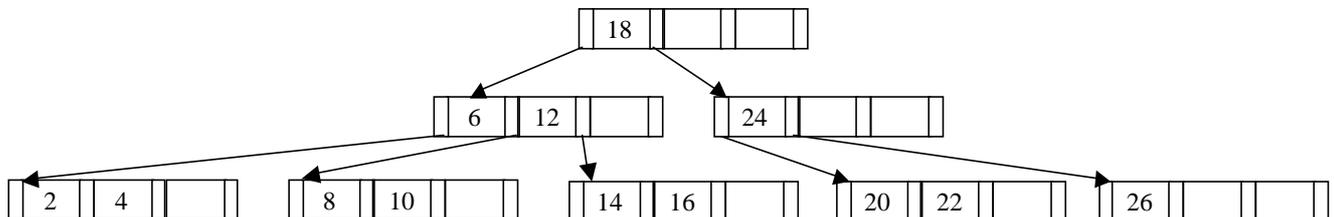
Inserción de 22:



Inserción de 24:



Inserción de 26:



Nota: Muéstrase al menos cada paso en la construcción del árbol en el que se añadan o eliminen nodos.

10)(2,5 puntos) Considérese un fichero con los campos A:Byte(2), B:String(10), C:String(20), con un índice hash sobre el campo A cuya función hash es $f(A)=A$, tamaño de bloque de 512 bytes, direcciones de bloque de 4 bytes y mapas de existencia. Se pide:

a. Calcular el espacio desperdiciado en un bloque en el mejor y peor casos.

- El mejor caso es con ocupación máxima del bloque. Para calcular el factor de bloqueo N (número de registros por bloque):

$$N*(2+10+20)*8+4*8+N \leq 512*8, \text{ donde:}$$



$N \cdot (2+10+20) \cdot 8 = N \cdot 256$ es el tamaño en bits ocupado por un registro
 $4 \cdot 8 = 32$ es el número de bits ocupado por la dirección del siguiente bloque
 N son los bits necesarios para el mapa de bits
 $512 \cdot 8 = 4096$ es el número de bits en los 512 bytes de un bloque
 $257 \cdot N \leq 4096 - 32$
 $N \leq 15,8$

Se toma $N=15$ registros por bloque. Por lo tanto, el espacio desperdiciado es:
 $4096 - 15 \cdot (2+10+20) \cdot 8 - 32 - 15 = 209$ bits, es decir, un poco más de 26 bytes (26,125 exactamente)

- El peor caso es con ocupación mínima del bloque, es decir, con sólo un registro:
 $4096 - (2+10+20) \cdot 8 - 32 - 15 = 3793$ bits, es decir, un poco más de 474 bytes (474,125 exactamente)
- b. Calcular el número de bloques de desbordamiento para 16.777.216 (16M) registros asumiendo una distribución uniforme.
 Para 16777216 registros con una distribución uniforme de valores de la clave de 0 a 65535 hay $16777216/65536=256$ registros por valor. Como en cada bloque sólo caben 15 registros, se necesitan $\lceil (256-15)/15 \rceil = 17$ bloques de desbordamiento por cada valor. En total se necesitan $17 \cdot 65536 = 1.114.112$ bloques de desbordamiento. Esto significa que en media se necesitarán $\lceil 17/2 \rceil = 9$ accesos de E/S para localizar un registro debido al acceso al bloque del fichero de datos en el que reside el registro.

- c. Calcular el número de bloques de desbordamiento para 16.777.216 (16M) registros asumiendo una distribución lineal $p(x)=mx$.
 En una distribución lineal $p(x)=mx$ de valores comprendidos entre 0 y 65.535 con 16.777.216 registros se cumple que (aproximación continua):

$$\int_0^{65535} mx dx = 16.777.216$$

$$\int_0^{65535} mx dx = \left[m \frac{x^2}{2} \right]_0^{65535} = \frac{m}{2} (65.535^2)$$

$$m = \frac{2 \cdot 16.777.216}{65.535^2} = 0,00781274$$

Los bloques de desbordamiento son necesarios a partir del valor de la clave x tal que $p(x)=mx=15$. Por tanto, $x = \lceil 1919,94 \rceil = 1920$ (para el valor de la clave 1919 hay 15 registros que caben en un solo bloque, y para el valor de la clave 1919 hay 16 registros). Por tanto, habrá los siguientes registros que habrá que ubicar en bloques de desbordamiento:

$$16.777.216 - 1920 - 15 \cdot (65.536 - 1920) = 15.821.056, \text{ donde}$$

El primer término, 16.777.216, son el total de registros.

El segundo término, 1920, son los primeros 1920 registros (valores de la clave comprendidos entre 0 y 1919) que caben en bloques del fichero de datos.

El tercer término, $15 \cdot (65.536 - 1920)$ son los siguientes registros a partir del valor de la clave 1920 que caben en bloques del fichero de datos.

Por tanto, serán necesarios $\left\lceil \frac{15.794.386}{15} \right\rceil = \lceil 1.054.737,07 \rceil = 1.054.738$ bloques de desbordamiento.

Aunque en este caso se necesita un número ligeramente inferior de bloques de desbordamiento con respecto al caso anterior, en media se necesitarán $1 + (1.054.738 / (65.536 - 1.920)) / 2 = 9,3$ accesos de E/S (despreciando los 1920 primeros) para localizar un registro. En ambos casos, se ve necesario cambiar la función hash.