

Temps: 2.30 h.

Notes: 6-07 tarda Revisió: 7-07 matí

Cada pregunta en un full separat, excepte la pregunta 1 que ha d'estar en dos fulls, un per a cada consulta.

1 (2 pt)

Disposem d'aquesta base de dades que enregistra la informació sobre despatxos i professors d'una universitat:

```
create table professors(  
  dni char(50),  
  nomProf char(50) unique,  
  telefon char(15),  
  primary key (dni));  
  
create table despatxos(  
  modul char(5),  
  numero char(5),  
  superficie integer not null check(superficie>12),  
  primary key (modul,numero));  
  
create table assignacions(  
  dni char(50),  
  modul char(5),  
  numero char(5),  
  instantInici integer,  
  instantFi integer,  
  primary key (dni, modul, numero, instantInici),  
  foreign key (dni) references professors,  
  foreign key (modul,numero) references despatxos)
```

Les assignacions amb valor nul com a instantFi són les que corresponen a l'instant actual. Un despatx, doncs, està ocupat si té alguna d'aquestes assignacions i està lliure si no en té cap.

Usant la clàusula DISTINCT només si és necessari, doneu una instrucció SQL que obtigui, sense repeticions:

a) (0.5 punts) Els professors que alguna vegada han estat assignats a un despatx de més de 16 metres quadrats. Concretament es vol obtenir els noms dels professors ordenats de manera ascendent.

b) (1.5 punts) Els mòduls amb més de 5 despatxos lliures. Concretament es vol obtenir el mòdul i la superfície lliure del mòdul.

2. (2 pt)

Donat el següent fragment d'una BD que guarda dades sobre una cadena de botigues:

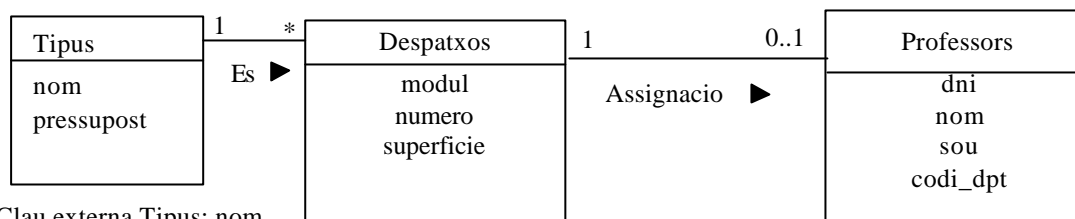
```
Botiga(idbotiga,nombotiga,adreça,telefon)  
Producte(idproducte,descripcio)  
Magatzem(idbotiga,idproducte,preu,estoc);  
{idbotiga} clau forana a botiga  
{idproducte} clau forana a producte
```

Es vol implementar mitjançant disparadors les següents restriccions.

a) L'estoc dels productes eliminats (per una instrucció DELETE) del magatzem no pot ser superior a l'estoc dels productes que queden.

b) Només podrà modificar les dades de la taula Botiga l'usuari amb el username "Joan".

3. (2 pt) Donat el disseny conceptual de la BD d'una universitat completament imaginària, on cada professor gaudeix d'un despatx individual d'un cert tipus amb un pressupost assignat per despeses i a més, hi ha un munt de despatxos lliures pendents d'assignació:



Clau externa Tipus: nom

Clau externa Despatxos: modul, numero

Clau externa Professors: dni

a) Transformeu aquest disseny conceptual a model relacional.

b) Donada la consulta SQL següent:

```

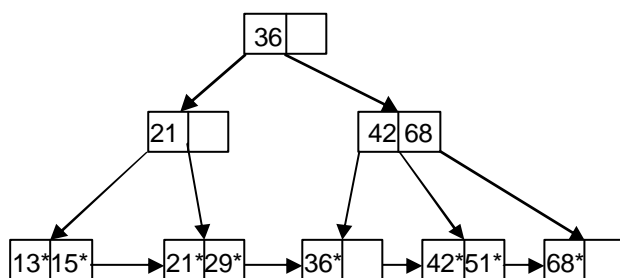
SELECT p.dni, p.nom, d.modul, d.numero, d.superficie
FROM Professors p, Despatxos d, Tipus t
WHERE p.modul=d.modul and p.numero=d.numero and d.nom=t.nom and
p.sou>t.pressupost and t.nom='petit';
  
```

Traduiu aquesta consulta SQL a Àlgebra Relacional.

c) Proposa l'arbre sintàctic òptim equivalent a la consulta de l'apartat anterior.

d) Supposeu ara que sabem que els professors que tenen un despatx del tipus 'petit' són sempre del departament d'LSI. Tenint present que encara que el departament de LSI és un dels més grans de la universitat, només té el 7% del total del professorat de la universitat. Amb aquesta nova informació seria possible introduir cap canvi a la consulta que permeti millorar el seu cost de resolució? En cas que la resposta sigui afirmativa, digues dins de quina/nes etapes del procés d'optimització classificaríes aquesta millora i proposa un nou arbre sintàctic òptim.

4. (2 pt) Considereu l'arbre B+ d'ordre 1



- A partir de la **figura inicial**, inseriu el 53* i doneu l'arbre resultant
- A partir de la **figura inicial**, elimineu el 36* i doneu l'arbre resultant
- Quin és el nombre màxim de registres que pot adreçar un arbre de 3 nivells i ordre 1? Per què?
- Quin és el nombre mínim de registres que pot adreçar un arbre de 3 nivells i ordre 1? Per què?

5 (2 pt.)

En una base de dades s'ha creat la taula T i s'han inserit les següent tuples mitjançant les sentències SQL següents :

```
Create table T(  
    a integer primary key,  
    b integer NOT NULL  
);  
insert into T values (1,1);  
insert into T values (2,2);  
insert into T values (3,3);
```

Tenim el següent horari :

T1	T2	T3
Select * from T where a=1 ;		
	update T set b=b+1 where a=1;	
		Select * from T;
Select * from T where a=1;		
COMMIT;		
	insert into T values (4,4);	
		Select * from T;
	COMMIT;	
		COMMIT;

- a) Suposant que es fa servir com a grànul la tupla, es produeixen interferències?Quines?
- b) Suposem que s'incorporen tècniques de reserva i que les reserves s'adquireixen a nivell de tupla. Les transaccions volen treballar amb SET TRANSACTION ISOLATION LEVEL REPEATABLE READ. Determinar i executar la seqüència d'operacions read i write que es generen, el seu ordre i les peticions de lock i unlock necessàries mostrant les esperes que s'hi produeixen.

Nota: Supposeu que per fer una inserció cal fer una lectura i una escriptura del grànul.

- c) Donar el graf d'espera corresponent a l'horari anterior abans del commit de T1.