

Løsningsforslag eksamen INF3100 våren 2015

Pasient(pnr, fnr, navn, adr)

Sykehusopphold(pnr, onr, inndato, utdato, avd, a dato)

Labprøve(prvnr, pnr, prdato, prtype, atype, resultat)

- I(i) pnr, onr \rightarrow inndato, utdato
pnr, inndato \rightarrow onr
pnr, a dato \rightarrow onr, avd

Kandidatnøkler: Alle må inneholde pnr, a dato siden disse ikke er i noen høyreside.

$(\text{pnr, a dato})^{\dagger} = \text{pnr, a dato, onr, avd, inndato, utdato}$

(pnr, a dato) er eneste kandidatnøkkel.

Normalformer:

pnr, onr \rightarrow inndato : vs ikke supernøkkel, så 7 BCNF
hs ikke nøkkelattributt, så 7 3NF
vs ikke ekte inneholdt i knk, så er 2NF

pnr, onr \rightarrow utdato : -||-

pnr, inndato \rightarrow onr : -||-

pnr, a dato \rightarrow onr : vs er supernøkkel, så er BCNF

pnr, a dato \rightarrow avd : -||-

Totalt er Sykehusopphold på 2NF, men bryter 3NF.

I (ii)

$(pnr, onr, inndato, utdato, avd, adato)$



$pnr, onr \rightarrow inndato$ bryter BCNF.
 $(pnr, onr)^+ = pnr, onr, inndato, utdato$

$(pnr, onr, inndato, utdato)$, $(pnr, onr, avd, adato)$

Lokale FDer:

$pnr, onr \rightarrow inndato$
 $pnr, onr \rightarrow utdato$
 $pnr, inndato \rightarrow onr$

Lokale knk:

(pnr, onr) og $(pnr, inndato)$

Så er på BCNF.

Lokale FDer:

$pnr, adato \rightarrow onr, avd$
Lokal knk: $pnr, adato$, så er på BCNF

De komposisjonen er FD-bevarende fordi ingen ikke-triviale FDer går på tvers av komponentene.

I (iii) Det er aldri mer enn ett resultat pr. analysetype:

$prvnr, atype \rightarrow resultat$

Et alternativt (like bra) svar er

$prvnr, prtype, atype \rightarrow resultat$

; men siden det er (mer eller mindre) underforstått at $prvnr \rightarrow prtype$, er det tilstrekkelig å si $prvnr, atype \rightarrow resultat$.

II (c)

Pasienter med flere opphold, pasientens navn uløst, ingen labtester tatt.

```
select p.pnr, count(distinct s.onr) as antopp  
from Pasient p, Sykehusoppd s  
where p.pnr = s.pnr and p.navn is null  
      and p.pnr not in (select pnr from Labprøve)  
group by p.pnr  
having count(distinct s.onr) > 1;
```

II (ii)

Oversikt over dagsbesøk:

```
create view Dagsbesøk as  
select *  
from Sykehusopphold  
where ihndato = utdato;
```

Dagsbesøk hvor det er tatt flere prøver:

```
create view Flereprøver as  
select d.pnr, d.onr  
from Dagsbesøk d, Labprøve p  
where d.pnr = p.pnr and !  
d.ihndato = p.prdato  
group by d.pnr, d.onr  
having count(distinct p.prvnr) > 1;
```

Dagsbesøk hvor det ikke er tatt noen prøver:

```
create view Ingenprøver as  
select pnr, onr  
from Dagsbesøk  
where (pnr, ihndato) not in  
(select pnr, prdato from Labprøve);
```

II (ii) (forts.)

Pasienter der dagsbesøkene har hatt nøyaktlig én prøvetaking pr. besøk:

```
create view AlltidEnPrøve as  
  select pnr          -- Bryr meg ikke med å ta distinct her.  
  from Dagsbesøk  
  where pnr not in (select pnr from Flereprøver) and  
        pnr not in (select pnr from Ingenprøver);
```

Pasienter blant disse med 5 eller flere dagsbesøk og der det har vært tatt samme prøvetype hver gang:

```
create view Sammeprøve5 as  
  select d.pnr  
  from AlltidEnPrøve a, Dagsbesøk d, Labprøve p  
  where a.pnr = d.pnr and d.pnr = p.pnr and d.inndato = p.inndato  
  group by d.pnr  
  having count(distinct d.pnr) >= 5 and  
        count(distinct p.prtype) = 1;
```

Det endelige svaret er da en opplisting av pasientnummer, prøvetype og analysetyper blant disse:

```
select distinct d.pnr, p.prtype, p.atype  
  from Sammeprøve5 s, Dagsbesøk d, Labprøve p  
  where s.pnr = d.pnr and d.pnr = p.pnr and d.inndato = p.prdato;
```

II (ii) En alternativ måte å finne pasienter der dagsbesøkene
(ants) har hatt nøyaktig én prøvetaking pr. besøk, er å bruke
outer join for å få med dagsbesøk uten prøvetakinger.
(count(distinct A) for et attributt A vil gi 0 hvis A
har null i alle tuplene; dette vil være tilfellet hvis
ingen prøver ble tatt på et dagsbesøk.)

create view AlltidEnPrøved as

select pnr

from (select Dagsbesøk.pnr, Dagsbesøk.onr

from Dagsbesøk left outer join Labprøve

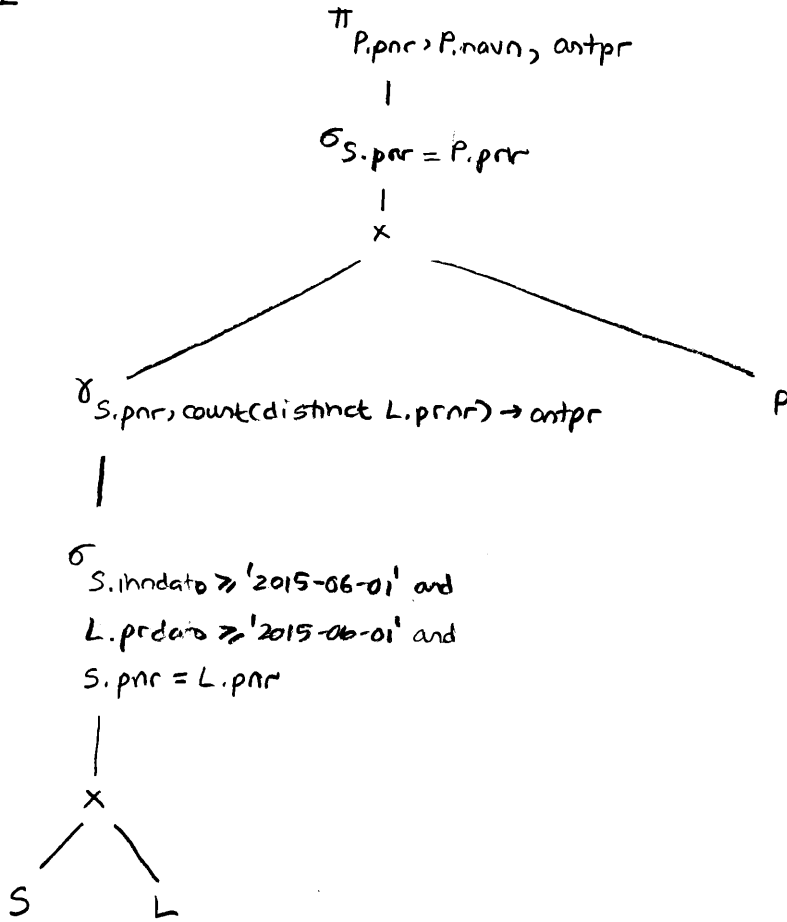
on Dagsbesøk.pnr = Labprøve.pnr and

Dagsbesøk.inndato = Labprøve.prdato

group by Dagsbesøk.pnr, Dagsbesøk.onr

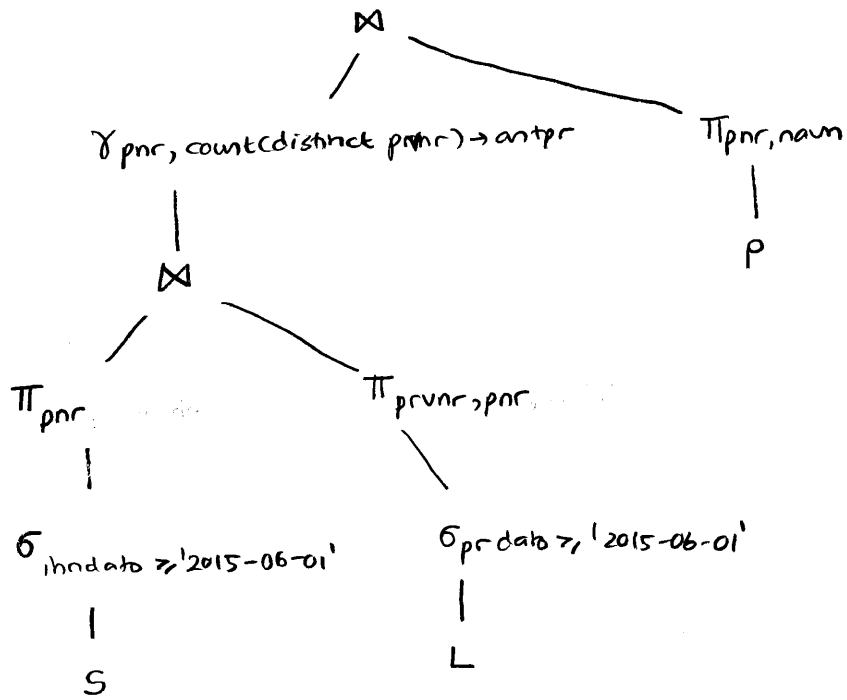
having count (distinct Labprøve.prvnr) = 1) as q;

III

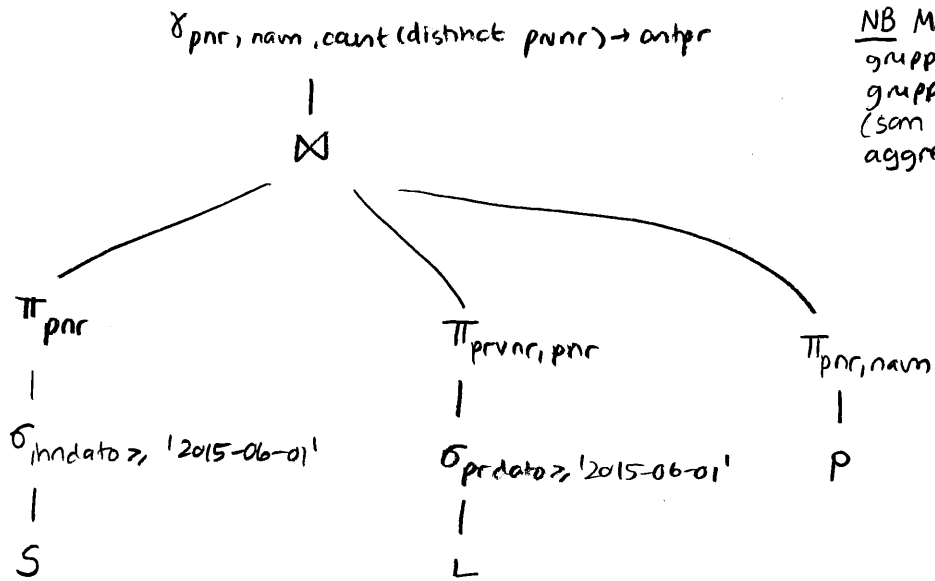


III (i) Spørsmålet finner antall prøver tatt av hver pasient med opphold etter 2015-06-01. (Antall prøver totalt, ikke fordelt på hvert enkelt opphold,)
(pr. pasient)

Förslag A:



Förslag B:



NB Må ha med navn i grupperingen, enten som grupperingsattribut (som i st), eller i en aggregering.

III (iii)

En pasient kan ikke ha to forskjellige opphold med samme utdato:

Forslag A:

$$\sigma_{\text{antut} > 1} (\gamma_{\text{pnr, onr, count}(\overset{\text{distinct}}{\text{utdato}})} \rightarrow \text{antut}(\text{Sykehusopp hold})) = \emptyset$$

Forslag B:

$$\rho_{S_1}(\text{Sykehusopp hold}) \bowtie \rho_{S_2}(\text{Sykehusopp hold}) = \emptyset$$

S1.pnr = S2.pnr
and S1.onr = S2.onr
and S1.utdato <> S2.utdato

Forslag B': (Kartesiske produkt + seleksjon i stedet for thetajoin)

$$\sigma_{\substack{S_1.pnr = S_2.pnr \\ \text{and} \\ S_1.onr = S_2.onr \\ \text{and} \\ S_1.utdato <> S_2.utdato}} (\rho_{S_1}(\text{Sykehusopp hold}) \times \rho_{S_2}(\text{Sykehusopp hold})) = \emptyset$$

IV

$S = r_1(a); r_1(c); w_1(a); r_2(a); r_2(c); w_2(c); r_1(b); w_1(b); c_1; c_2$

IV (i)



Siden grafen er uten syklor, er S konfliktserialiserbar.

IV (ii)

$T_1: l_1(a); r_1(a); l_1(c); r_1(c); w_1(a); l_1(b); r_1(b); w_1(b); c_1; u_1(a); u_1(c); u_1(b)$

$T_2: l_2(a); r_2(a); l_2(c); r_2(c); w_2(c); c_2; u_2(a); u_2(c)$

IV (iii)

S kan ikke genereres av en strikt 2PL-planlegger fordi T_2 ikke kan få lås på a før denne slippes av T_1 , og det skjer ikke før etter at T_1 har committet.

IV (iv)

Nei, det finnes ingen ikke-seriell plan for T_1 og T_2 som kan genereres av en strikt 2PL-planlegger; hvis T_1 får første lås (på a), hindres T_2 i å få sin første lås (på a) før T_1 er ferdig. Det samme er tilfellet hvis T_2 får første lås før T_1 . Så eneste mulige strikte 2PL-planer er $T_1; T_2$ og $T_2; T_1$ (altså serielle).

V

- (i) I RAID5 tåles 1 diskkræsje, mens 2 diskkræsje gir varig tap av data. I RAID51 er det derfor slik at hvis 2 disk og deres speilinger kræsjer, så går data tapt. Så det er mulig å tape data ved 4 samtidige diskkræsje. Ved 3 samtidige diskkræsje har én av gruppene maksimalt 1 diskkræsje, så da går ingen data tapt. Så minste antall kræsje som kan gi varig tap av data, er 4.
- (ii) Hvis samtlige disk i én gruppe kræsjer, og én disk i den andre, så kan data fortsatt gjenopprettes. Men samtlige i den ene gruppen og to i den andre gir varig tap av data. Så det største antall disk som kan kræsje samtidig uten varig tap av data, er $n+1$. (Mer generelt er at vi har $n-1$ "forskjellige" disk innlagt på tvers av de to gruppene, da kan vi også gjenopprette data. Da er det også maksimalt $n+1$ diskkræsje.)