

# INF3100 - Databasesystemer

## Ekstraoppgaver

Institutt for Informatikk  
Universitetet i Oslo

Mars 2014

### Løsningsforslag

Løsningsforslag fra khien-kin  
med kommentarer fra Ellen.

Bruker konsekvent bag-  
varianten av relasjonsalgebraen.  
Det betyr bl.a. at -, U, n er  
bagminus, bagunion, bagsnitt.

Professor(ssn, profname, status, salary)  
Course(crscod, crsname, credits)  
Taught(crscod, semester, ssn)

### Oppgave 1

Finn de ansatte som har forelest kurset 'INF1300', men har aldri forelest 'INF3100'. (Kunne ha brukt *not in*, *not exists* enn *except*).

#### RelasjonsAlgebra

$$\delta(\pi_{ssn}(\sigma_{crscod='INF1300'}(\text{Taught}))) - \pi_{ssn}(\sigma_{crscod='INF3100'}(\text{Taught}))$$

#### SQL

```
select ssn
from Taught
where crscod = 'INF1300'
except (
select ssn
from Taught
where crscod = 'INF3100');
```

Her er det mengde-  
différanse vi må ha;  
det får vi til ved å  
gjøre første argument  
til en mengde og bruke  
bagdifféranse (-).

## Oppgave 2

Finn de ansatte som har forelest både 'INF1300' og 'INF3100'.

### RelasjonsAlgebra

$$\delta(\pi_{ssn}(\sigma_{crscode='INF1300'}(\text{Taught})) \cap \pi_{ssn}(\sigma_{crscode='INF3100'}(\text{Taught})))$$

Kan først bruke  
bagsnitt og deretter  
fjerne flerforekomster.  
(Dette går bra for  
snitt, men ikke for  
differanse.)

### SQL

#### Løsning 1

```
select ssn
from Taught
where crscode = 'INF1300'
intersect (
select ssn
from Taught
where crscode = 'INF3100');
```

#### Løsning 2

```
select t1.ssn
from Taught t1, Taught t2
where t1.crscode = 'INF1300' and t2.crscode = 'INF3100';
```

and t1.ssn = t2.ssn

## Oppgave 3

Finn de ansatte som aldri har forelest 'INF3100'

### RelasjonsAlgebra

$$\pi_{ssn}(\text{Professor}) - \pi_{ssn}(\sigma_{crscode='INF3100'}(\text{Taught}))$$

### SQL

```
select ssn
from Professor
except(
select ssn
from Taught
where crscode = 'INF3100');
```

Eller except all fordi select ssn from Professor  
ul returnere en mengde (siden ssn er primærnøkkel  
i Professor).

....(Taught t...) er feil notasjon,  
 Rett notasjon er  $\rho_t(\text{Taught})$ , dvs. omnavning.

## Oppgave 4

Finn de ansatte som har forelest både 'INF1000' og 'INF1010' på samme semesteret.

### RelasjonsAlgebra

$$\delta \left( \pi_{t1.ssn}(\sigma_{t1.crscode='INF1000' \wedge t2.crscode='INF1010' \wedge t1.semester=t2.semester}(\text{Taught } t1 \times \text{Taught } t2)) \right)$$

$\wedge t1.ssn = t2.ssn$

$\rho_{t1}(\text{Taught})$

### SQL

```

distinct
select t1.ssn
from Taught t1, Taught t2
where t1.crscode = 'INF1000' and t2.crscode = 'INF1010'
and t1.semester = t2.semester;

```

Eller:

$$\delta \left( \pi_{t1.ssn} \left( \rho_{t1}(\text{Taught}) \bowtie \rho_{t2}(\text{Taught}) \right) \right)$$

$t1.semester = t2.semester \wedge$   
 $t1.ssn = t2.ssn \wedge$   
 $t1.crscode = 'INF1000' \wedge$   
 $t2.crscode = 'INF1010'$

## Oppgave 5

Finn de professorene som har forelest 'INF3190' eller 'INF1060', men ikke undervist begge.

### RelasjonsAlgebra

$$\delta \left( \pi_{ssn}(\sigma_{crscode \neq 'INF3190' \vee crscode \neq 'INF1060'}(\text{Taught})) \right) - \left( \pi_{ssn}(\sigma_{crscode='INF3190'}(\text{Taught})) \cap \pi_{ssn}(\sigma_{crscode='INF1060'}(\text{Taught})) \right)$$

### SQL

```

select ssn
from Taught =
where crscode \neq 'INF3190' OR
crscode = 'INF1060'
except (
select ssn
from Taught
where crscode = 'INF3190'
intersect (
select ssn
from Taught

```

```
where crscode = 'INF1060'));
```

## Oppgave 6

Finn de kursene som har aldri blitt forelest

## RelasjonsAlgebra

$$\pi_{crscode}(\text{Course}) - \pi_{crscode}(\text{Taught})$$

## SQL

```
select crscode
from Course
except(
select crscode
from Taught);
```

## Oppgave 7

Finn de kursene som har blitt undervist i minst to semester.

## RelasjonsAlgebra

$$\delta(\pi_{t1.crscode}(\sigma_{t1.semester <> t2.semester \wedge t1.crscode = t2.crscode}(\rho_{t1}(\text{Taught}) \times \rho_{t2}(\text{Taught}))))$$

Eller:  
 $\delta(\pi_{t1.crscode}(\rho_{t1}(\text{Taught}) \bowtie \rho_{t2}(\text{Taught})))$   
 $t1.crscode = t2.crscode \wedge$   
 $t1.semester <> t2.semester$

## SQL

```
distinct
select t1.crscode
from Taught t1, Taught t2
where t1.semester <> t2.semester and
t1.crscode = t2.crscode;
```

## Oppgave 8

Finn de kursene som har blitt undervist i minst ti semester.

## RelasjonsAlgebra

~~$\sigma_{count(*) \geq 10}(\gamma_{crscode}(\text{Taught}))$~~

NEI! Feil notasjon - kan ikke ha aggregering i  $\sigma$ !  
Bare gruppering ( $\gamma$ ) kan ha aggregering!

4

$$\pi_{crscode}(\sigma_{sc \geq 10}(\gamma_{crscode, count(semester) \rightarrow sc}(\text{Taught})))$$

## SQL

```
select crscode
from Taught
group by crscode
having count(*) >= 10
```

## Oppgave 9

Finne de kursene som har blitt undervist av fem forskjellige ansatte.

## RelasjonsAlgebra

Let's create a view:

NEI!  
~~UniqueCourses  $\leftarrow \delta_{crscode, ssn}(\text{Taught})$~~   
 ~~$\sigma_{count(*) \geq 5}(\gamma_{crscode}(\text{UniqueCourses}))$~~

## SQL

```
create view UniqueCourses as (
select distinct crscode, ssn
from Taught);
```

```
select crscode
from UniqueCourses
group by crscode
having count(*) = 5;
```

Eller: select crscode  
from Taught  
group by crscode  
having count(distinct ssn) = 5;

## Oppgave 10

Finne de professorene som i det hele tatt har undervist 'INF5050'.

## RelasjonsAlgebra

$\delta(\pi_{profname}(\sigma_{crscode='INF5050'}(\text{Taught} \bowtie \text{Professor})))$

## SQL

```
select distinct profname
from Taught natural join Professor
```

Med "view"-notasjon i relasjonsalgebraen skal svaret alltid kalles 'Answer'

UniqueCourses  $\leftarrow \delta(\pi_{crscode, ssn}(\text{Taught}))$

Answer  $\leftarrow \pi_{crscode}(\sigma_{n=5}(\gamma_{crscode}(\text{UniqueCourses})))$

$\delta_{crscode, count(ssn) \rightarrow n}(\text{UniqueCourses})$

where crscode = 'INF5050';

## Oppgave 11

Finn de Fullprofessorene som i det hele tatt har undervist 'INF1000'.

### RelasjonsAlgebra

$\delta(\pi_{\text{profname}}(\sigma_{\text{crscode}='INF1000' \wedge \text{status}='Full'}(\text{Taught} \bowtie \text{Professor})))$

### SQL

*distinct*  
select profname  
from Taught natural join Professor  
where crscode = 'INF1000' and status = 'Full';

## Oppgave 12

Finn de professorene som har undervist i mer enn to kurs iløpet av et semester.

### RelasjonsAlgebra

NEI!

~~$\text{TaughtTwoSemester} \leftarrow (\sigma_{\text{count}(*), > 2}(\gamma_{\text{ssn}, \text{semester}}(\text{Taught})))$~~

~~$\pi_{p.\text{profname}}(\sigma_{p.\text{ssn}=t.\text{ssn}}(\text{Professor } p \times \text{TaughtTwoSemester } t))$~~

### SQL

select profname  
from Professor  
where ssn in(  
select ssn  
from Taught  
group by ssn, semester  
having count(\*) > 2);

$\text{TaughtTwo} \leftarrow \pi_{\text{ssn}}(\sigma_{n > 2}(\gamma_{\text{ssn}, \text{semester}, \text{count}(*), n}(\text{Taught})))$

$\text{Answer} \leftarrow \pi_{\text{profname}}(\text{Professor} \bowtie \text{TaughtTwo})$

### Oppgave 13

Fjern de professorene som har aldri undervist noe i det hele tatt.

```
delete from Professor
where ssn not in
(select ssn
from Taught);
```

### Oppgave 14

Skift studiepoeng til 10 for de kursene som har vært i Høsten 2003.

```
update Course
set credits = 10
where crscode in (
select crscode
from Taught
where semester = 'H03');
```

### Oppgave 15

Finn de professorene som har undervist i fag som har gitt 20 studiepoeng.

#### RelasjonsAlgebra

$$\delta_c(\pi_{p.\text{profname}}(\sigma_{p.\text{ssn}=t.\text{ssn} \wedge t.\text{crscode}=c.\text{crscode} \wedge c.\text{credits}=20}(\rho_p(\text{Professor}) \times \rho_t(\text{Taught}) \times \rho_c(\text{Course}))))$$

Eller:  $\delta(\pi_{\text{profname}}(\pi_{\text{ssn}}(\sigma_{\text{credits}=20}(\text{Course} \bowtie \text{Taught}))) \bowtie \text{Professor}))$

#### SQL

```
select distinct p.profname
from Professor p, Taught t, Course c
where p.ssn = t.ssn and t.crscode = c.crscode and c.credits = 20);
```

### Oppgave 16

Finn de professorene som har undervist mest antall kurs i semesteret Vår 2012.

NB! Oppgave 16 ønsker vi å aggregere over en hel relasjon. Da er det underforstått en gruppering med null grupperingsattributter! ⊕

$AntKurs12 \leftarrow \gamma_{ssn, count(crscode) \rightarrow nc} (\sigma_{semester='V12'}(Taught))$

$MaksAntKurs \leftarrow \gamma_{max(nc) \rightarrow m} (AntKurs12)$

$Answer \leftarrow \pi_{a1, ssn} (\rho_{a1} (AntKurs12) \bowtie \rho_{a2} (MaksAntKurs))$

$a1.nc = a2.m$

## RelasjonsAlgebra

Vet ikke om dette er 100% korrekt.

~~$TotalCourseLectured \leftarrow (\gamma_{ssn, count(*) \rightarrow num} (\sigma_{semester='V12'}(Taught)))$~~

~~$MostLecturedProfessor \leftarrow (\gamma_{ssn} (\sigma_{semester='V12'}(Taught)))$~~

~~$\pi_{p, profname} (\sigma_{p.ssn=m.ssn} (Professor p \times MostLecturedProfessor m))$~~

## SQL

```
select profname
from Professor
where ssn in (
select ssn
from Taught
where semester = 'V12'
group by ssn
having count(*) = (select max(num)
from (select ssn, count(*) as num
from Taught
where semester = 'v12'
group by ssn)
));
```

## Oppgave 17

List opp alle professorene som har 'Smith' som en del av navnet og har undervist i semesteret Høst 2011.

## RelasjonsAlgebra

$\pi_{crsname} (\sigma_{profname \text{ like } \%Smith\% \wedge semester='H11'} (Professor \bowtie Taught \bowtie Course))$

## SQL

```
select crsname
from Professor natural join Taught natural join Course
where profname like '%Smith%' and semester = 'H11';
```

⊕ Et alternativ til å gruppere med null grupperingsattributter, er å innføre et kunstig ekstra attributt hvor alle tuplere får lik verdi, og så aggregere på dette. Det kan vi få til med utvidet projeksjon:

$MaksAntKurs \leftarrow \pi_m (\gamma_{extra, max(nc) \rightarrow m} (\pi_{nc, 1 \rightarrow extra} (AntKurs12)))$



## Oppgave 18

List opp antall kurs i sortert rekkefølge hvor professor med ssn = 123456789 har undervist i dette semesteret.

### RelasjonsAlgebra

$\tau_{semester\ asc}(\gamma_{semester, count(*)}(\sigma_{ssn=123456789}(Taught)))$

SQL  $\tau_{semester\ asc}(\gamma_{semester, count(*)} \rightarrow nc(\sigma_{ssn=123456789}(Taught)))$

```
select semester, count(*)
from Taught
where ssn = 123456789
group by semester
order by semester asc;
```

## Oppgave 19

List opp i alfabetisk rekkefølge over navnt på professorene og antall kurs han/hun har undervist.

### RelasjonsAlgebra

$\rho_p(Professor) \rho_t(Taught)$

$\tau_{p.profname\ asc}(\gamma_{p.ssn, p.profname, count(*)}(\sigma_{p.ssn=t.ssn}(Professor \times Taught)))$

SQL Eller:  $\tau_{profname}(\gamma_{ssn, profname, count(course)} \rightarrow nc(Professor \bowtie Taught))$

```
select p.profname, count(*)
from Professor p, Taught t
where p.ssn = t.ssn
group by p.ssn, p.profname
order by p.profname asc;
```

## Oppgave 20

Finn de professorene som ikke har undervist iløpet av semesteret Vår 2010, Høst 2010 og Vår 2011. (RelasjonsAlgebra, kunne vi funnet med differanse eller del på).

$\pi_{ssn}(\text{Professor})$

-  
 $\pi_{ssn}(\sigma_{\text{semester}='V10' \vee \text{semester}='H10' \vee \text{semester}='V11'}(\text{Taught}))$

## SQL

```
select *
from Professor p
where not exists (
select *
from Taught t
where p.ssn = t.ssn and (t.semester = 'V10' or t.semester = 'H10' or
t.semester = 'V11'));
```

## Oppgave 21

Finn kursene som 'Thomas Plagemann' har aldri undervist før.

## RelasjonsAlgebra

$\pi_{crsname}(\text{Course}) - \pi_{crsname}(\sigma_{\text{profname}='ThomasPlagemann'}(\text{Professor} \bowtie \text{Taught} \bowtie \text{Course}))$

## SQL

```
select crsname
from Course c
where not exists (
select *
from Professor p, Taught t
where p.profname = 'Smith' and p.ssn = t.ssn and
t.crscode = c.crscode);
```

## Oppgave 22

Finn alle kursene som har blitt undervist av alle professorene på ifi.

## RelasjonsAlgebra

$\delta(\pi_{crscode,ssn}(\text{Taught})) / \delta(\pi_{ssn}(\text{Professor}))$

## SQL

```
select crscode
from Taught T1
```

(Strengt tatt har vi ikke definert divisjon for bagvarianten av relasjonsalgebra, men hvis vi er sikre på at begge argumentene er mengder, går det i hvertfall bra.)

```

where not exists(
    (select ssn
     from Professor)
except
    (select ssn
     from Taught T2
     where T2.crscode = T1.crscode));

```

## Oppgave 23

Finn alle kursene som har blitt undervist i alle semesteret som fins.

### RelasjonsAlgebra

$$\delta(\pi_{crscode,semester}(Taught) / \delta(\pi_{semester}(Taught)))$$

### SQL

```

select crscode
from Taught T1
where not exists(
    (select semester
     from Taught)
except
    (select semester
     from Taught T2
     where T2.crscode = T1.crscode));

```

## Oppgave 24

Finn de kursene som bare har blitt undervist av Phd studenter.

### RelasjonsAlgebra

$$\pi_{crscode}(Course) - \pi_{crscode}(\sigma_{status \neq 'Phd'}(Professor \bowtie Taught))$$

### SQL

```

select crscode

```

```

from Course C
where c.crscod not in(
    (select crscod
     from Taught T, Professor P
     where T.ssn = P.ssn and P.status != 'Phd');

```

## Oppgave 25

Finn de professorene som har undervist det meste antall kurs iløpet av semester Vår 2006.

(Se oppgave 16)

## SQL

```

select *
from Professor P1
where not exists
(
    select *
    from Professor P2
    where(
        (select count(*)
         from Taught
         where Taught.ssn = P2.ssn and Taught.semester='V06')
        >
        (select count(*)
         from Taught
         where Taught.ssn = P1.ssn and Taught.semester='V06'));

```

## Oppgave 26

Finn de professorene som tjener mest av alle.

## RelasjonsAlgebra

$HighestSalary \leftarrow (\pi_{max(salary) \rightarrow highSal}(Professor))$   
 $\pi_*(\sigma_{salary=highSal})(Professor \times HighestSalary)$

NEI!  
Meningsløst å  
prosjekt på "alt".

NEI! Kan ikke aggregere i  $\pi$

NB! Intet grupperingsattributt:  
Aggregert over hele relasjonen Professor

## SQL

```

select *
from Professor

```

$HighestSalary \leftarrow \rho_{max(salary) \rightarrow highSal}(Professor)$

$Answer \leftarrow \pi_{ssn, profname}(Professor \bowtie_{salary=highSal} HighestSalary)$

```
where salary = (select max(salary)
from professor);
```

## Oppgave 27

Finn de professorene som tjener nest mest av alle.

## RelasjonsAlgebra

~~$\pi_*(\sigma_{1=(\pi_{count(*)}(\sigma_{p2.salary > p1.salary}(Professor p2)))}(Professor p1))$~~

NEI! Kan ikke  
ha relalguttrykk som  
argument til f.eks.  
betingelser i  $\sigma$

## SQL

```
select *
from Professor p1
where 1 = (select count(*)
from Professor p2
where p2.salary > p1.salary);
```

$MaxSalary \leftarrow \gamma_{max(salary) \rightarrow ms} (Professor)$

$NotMaxSalary \leftarrow \delta(\pi_{salary} (Professor)) - \pi_{ms} (MaxSalary)$

$SecMaxSalary \leftarrow \gamma_{max(salary) \rightarrow nms} (NotMaxSalary)$

$Answer \leftarrow \pi_{ssn} (Professor \bowtie_{salary = nms} SecMaxSalary)$

Her skal vel strengt tatt argumentet  
'ms' omnavnes til 'salary':

$\rho_{MaxS(salary)} (\pi_{ms} (MaxSalary))$