Løsningsforslag eksamen Inf3100 våren 2009 (bortsett fra oppgave 3b som skal løses som del av oblig2 våren 2011

Oppg.1

1a

(i) fødselsNr → søknadsNr, navn, karaktersnitt
søknadsNr → fødselsNr
} i Elev

søknadsNr →→ skole
(søknadsNr →→ linje – unødvendig, for følger av søknadsNr →→ skole)
} i Søknad

(ii) Kandidatnøkler i Elev: Siden

fødselsNr$^+$ = fødselsNr, søknadsNr, navn, karaktersnitt
søknadsNr$^+$ = -"-

er hver av {fødselsNr} og {søknadsNr} kandidatnøkler.
(Dette er de eneste; de øvrige alternativene forekommer bare i høyresider)

Da begge venstresidene i de to FDene i Elev er kandidatnøkler, er de også supernøkler, og oppfyller følgelig kravene til BCNF.
(Det finnes ingen ikketrivielle MVDer i Elev, så 4NF er også oppfylt.)

(iii) Kandidatnøkler i Søknad:

Det er ingen ikketrivielle FDer i Søknad, så alle alternativene {søknadsNr, skole, linje} danner en kandidatnøkkel (den eneste).

MVDen er ikketrivell (høyresiden ikke del av venstresiden, vs + hs ikke alle attr.)

Venstresiden i søknadsNr →→ skole er ikke en kandidatnøkkel og altså ikke en supernøkkel, så det er brudd på 4NF. Derimot er BCNF oppfylt.

1b

(i)
```
select s.skole, l.linje, count(s.søknadsNr) as antallsøkere
from Skolesøknad s, Linjesøknad l
where s.søknadsNr = l.søknadsNr
group by s.skole, l.linje
order by antallsøkere desc;
```

(ii)
```
select
from Tilbud t
where (t.skole, t.linje) not in (select s.skole, l.linje
                                 from Skolesøknad s, Linjesøknad l
                                 where s.søknadsNr = l.søknadsNr);
```

→ De skole-linje-kombinasjonene som det finns søknader til

ELLER F.EKS.

```
(select skole, linje
from Tilbud)
except all
(select skole, linje from (Skolesøknad natural join Linjesøknad));
```

→ Kan bruke except all fordi første select er en mengde siden (skole, linje) er primærnøkkel.

1b

(iii)

```
select
from Elev e
where not exists ( (select s.skole, l.linje
                    from Skolesøknad s, Linjesøknad l, Søknad a,
                    where a.fødselsNr = e.fødselsNr and
                          a.søknadsNr = s.søknadsNr and
                          a.søknadsNr = l.søknadsNr)
                   intersect
                   (select skole, linje
                    from Tilbud));
```

Skulle gå bra med intersect all også, for select skole, linje returnerer en mengde

De skole-linje-kombinasjonene som tilbys

De skole-linje-kombinasjonene som e har søkt

1c

$$\tau_{\text{antallsøkere desc}}\left(\gamma_{\text{skole, linje, count(søknadsNr)}\rightarrow\text{antallsøkere}}\left(\text{Skolesøknad} \bowtie \text{Linjesøknad}\right)\right)$$

Kan eventuelt ha med $\pi_{\text{skole, linje, antallsøkere}}$ her, men det er unødvendig fordi det er disse som uansett leveres i svaret fra γ.

Oppgave 2

(i) En tapsfri dekomposisjon er en oppsplitting i relasjoner der oppsplittingen er slik at det ikke dannes falske tupler ved naturlig join.

Litt mer presist: En tapsfri dekomposisjon av en relasjon R er en samling relasjoner $R_1, \ldots, R_n$ hvor vi for enhver lovlig instans av R har at hvis instansen projiseres på (attributtene i) $R_1, \ldots, R_n$ og vi deretter beregner en naturlig join av projeksjonene igjen, så gjenskapes den opprinnelige instansen nøyaktig.

(ii)

	A	B	C	D	E	F	G	H
ABCE	a	b	c	$\not{d}_1$ d	e	$\not{f}_1$ f	$\not{g}_1$ g	$\not{h}_1$ h
ACFGH	a	b_2	c	$\not{d}_2$ d	$\not{e}_2$ e	f	g	h
BCDFG	$\not{a}_3$ a	b	c	d	$\not{e}_3\not{e}_2$ e	f	g	$\not{h}_3$ h
BDEG	$\not{a}_4$ a	b	c_4	d	e	$\not{f}_4\not{f}_1$ f	g	h_4

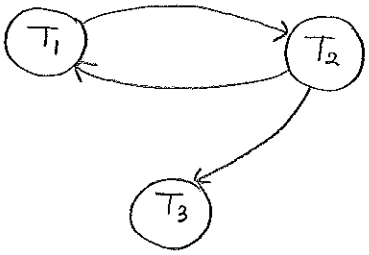
Bruker Chasealgoritmen med de oppgitte FDene. Siden minst én rad da får kun bokstaver uten indekser, er dekomposisjonen tapsfri.

(Det er ikke nødvendig å fortsette chasen etter at en slik rad har oppstått; f.eks. kunne vi ha stanset på et tidspunkt der ay fortsatt var en verdi, men hver rad i alt var uten indekser.)

Oppgave 3

3a

(i)



Siden presedensgrafen har en sykel, er S ikke konfliktserialiserbar.

($T_1 \to T_2$ fordi f.eks. $w_1(b) <_S r_2(b)$
$T_2 \to T_1$ -"- $w_2(c) <_S r_1(c)$
$T_2 \to T_3$ -"- $w_2(c) <_S r_3(c)$)

(ii)

T_1	T_2	T_3
$l_1(a)$		
$r_1(a)$		
$l_1(b)$		
$r_1(b)$		
	$l_2(c)$	
	$r_2(c)$	
$w_1(b)$		
	$l_2(b)$ - vent på T_1	
		$l_3(d)$
		$r_3(d)$
$l_1(c)$ - vent på T_2; vrangläs mot T_2		
		$l_3(c)$ - vent på T_2; T_3 blir også rammet av vrangläsen mellom T_1 og T_2

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

T_2: $l_2(c)$; $r_2(c)$; $l_2(b)$; $r_2(b)$; $w_2(c)$; $u_2(c)$; $w_2(b)$; $u_2(b)$

T_3: $l_3(d)$; $r_3(d)$; $l_3(c)$; $r_3(c)$; $w_3(c)$; $u_3(c)$; $w_3(d)$; $u_3(d)$]