

UNIVERSITETET I OSLO

Det matematisk-naturvitenskapelige fakultet

Eksamen i INF3100 — Databasesystemer

Eksamensdag: 9. juni 2008

Tid for eksamen: 14.30–17.30

Oppgavesettet er på 5 sider.

Vedlegg: Ingen

Tillatte hjelpemidler: Kalkulator og ordbok

Kontroller at oppgavesettet er komplett før
du begynner å besvare spørsmålene.

Les oppgavene nøye, og lykke til!

Oppgave 1 Relasjonsdatabaser (40%)

Det begrepsmessige skjemaet for en relasjonsdatabase for plassreservering i tog inneholder (blant annet) følgende tabeller:

```
Tog(togNr, startSt, endeSt, ankomstTid)
TogTabell(togNr, stasjon, avgangstid)
Plass(dato, togNr, vognNr, plassNr, vindu, ledig)
```

- togNr, vognNr og plassNr har domene INTEGER
- startSt, endeSt og stasjon har domene VARCHAR(30)
- vindu og ledig har domene BOOLEAN
- avgangstid og ankomstTid har domene TIME
- dato har domene DATE
- Tog har primærnøkkel togNr
- TogTabell har primærnøkkel (togNr, avgangstid)
- Plass har primærnøkkel (dato, togNr, vognNr, plassNr)
- togNr er fremmednøkkel til Tog både i TogTabell og Plass
- Plass har følgende FD: (togNr, vognNr, plassNr) → vindu

(Fortsettes på side 2.)

1a (10%)

Hvilken normalform har hver av de tre relasjonene?
Begrunn svarene.

1b (10%)

NSB vurderer å splitte relasjonen `Plass` i en ny relasjon `Sete` og en endret relasjon `Plass` slik at datastrukturen blir:

```
Tog(togNr, startSt, endeSt, ankomstTid)
TogTabell(togNr, stasjon, avgangstid)
Sete(togNr, vognNr, plassNr, vindu)
Plass(dato, togNr, vognNr, plassNr, ledig)
```

Er denne dekomposisjonen av `Plass` tapsfri?
Hvilken normalform har `Sete` og `Plass`?
Ser du noen problemer med denne dekomposisjonen?
Begrunn svarene.

1c (10%)

Anta at NSB velger datastrukturen i deloppgave 1b. Bruk relasjonsalgebra til å finne `vognNr` på de vognene i tog nr. 401 som den 10.6.2008 ikke har noen ledige vindusplasser.

1d (10%)

Uttrykk i relasjonsalgebra at `(togNr, stasjon)` er kandidatnøkkel i relasjonen `TogTabell`.

Hint: Husk at vi i relasjonsalgebraen kan ha booleske uttrykk av formen $E_1 \subseteq E_2$ der E_1 og E_2 er relasjonsuttrykk.

(Fortsettes på side 3.)

Oppgave 2 ODL/OQL (20%)

Vi skal se på en liten database for et firma som driver kursvirksomhet. Kursene har entydige navn, og de samme kursene blir holdt hvert år. Kursdeltakerne kalles kunder. Her følger ODL-skjema for databasen:

```
class Kunde (extent kunder key (e-post))
{ attribute    integer mobil;
  attribute    String  e-post;
  attribute    String  navn;
  attribute    Date    f-dato;
  relationship set<Kurs> tar_kurs
    inverse Kurs::deltakere;
};
class Kurs (extent kursene key (kursnavn, aar))
{ attribute    String  kursnavn;
  attribute    integer aar;
  relationship set<Kunde> deltakere
    inverse Kunde::tar_kurs;
};
```

2a (5%)

Bruk OQL til å lage en liste over kursene i 2007 og antall deltakere på hvert av dem. Sorter listen synkende etter antall deltakere.

2b (10%)

Finn navn og e-postadresse for kunder som har tatt alle kursene firmaet tilbyr (kundene kan ha brukt flere år på dette).

2c (5%)

Forklar kort (uten SQL-kode) hvordan denne datastrukturen hadde sett ut hvis vi hadde valgt å bruke en relasjonsdatabase. Det er nok å oppgi navn på relasjoner og attributter, og å si hvilke kandidatnøkler og fremmednøkler hver relasjon har.

(Fortsettes på side 4.)

Oppgave 3 Transaksjonshåndtering (30%)

Anta at vi bare bruker binære (eksklusive) låser. Vi skriver $l_i(x)$ og $u_i(x)$ for å betegne at transaksjonen t_i henholdsvis setter eller frigir en lås på dataelementet x . Som vanlig betyr $r_i(x)$ at t_i leser x og $w_i(x)$ at t_i skriver x . Betrakt følgende eksekveringsplan (schedule) for de tre transaksjonene t_1 , t_2 og t_3 :

$$S_1 : \quad l_1(x)r_1(x)l_2(y)r_2(y)l_3(z)r_3(z)w_2(y)u_2(y)w_1(x)u_1(x)w_3(z)u_3(z) \\ l_3(y)r_3(y)l_2(x)r_2(x)l_1(z)r_1(z)w_2(x)u_2(x)w_3(y)u_3(y)w_1(z)u_1(z)$$

3a (5%)

Tegn presedensgrafene for S_1 og avgjør om S_1 er konfliktserialiserbar.

3b (10%)

Begrunn at S_1 ikke tilfredsstillere 2-faselåsingsprotokollen (2PL). Lag en ny plan for de tre transaksjonene som tilfredsstillere 2PL. Vis at den nye planen er serialiserbar.

3c (10%)

Betrakt så følgende eksekveringsplan for de tre transaksjonene t_4 , t_5 og t_6 :

$$S_2 : \quad r_4(x)w_4(x)r_5(x)r_6(y)w_5(x)w_5(y)w_6(x)$$

Tegn polygrafene og avgjør om S_2 er view-serialiserbar.

3d (5%)

Er S_2 konfliktserialiserbar? Begrunn svaret.

(Fortsettes på side 5.)

Oppgave 4 Moderne RAID-teknologi (10%)

Stadig billigere diskene gjør det mulig med nye metoder for å redusere risikoen for at diskkrasj medfører at hele systemet går ned. En slik metode er å kombinere RAID 5 med RAID 1, dvs. at vi organiserer en gruppe diskene som RAID 5 (paritetsblokker fordelt på alle diskene i gruppen), og så speiler vi hele gruppen. Denne metoden kalles RAID 51.

Hva er det minste antall diskene som må krasje samtidig for at systemet skal gå ned når vi bruker RAID 51? Begrunn svaret.